

**KOCAELİ ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**BİLGİSAYAR MÜHENDİSLİĞİ  
ANABİLİM DALI**

**YÜKSEK LİSANS TEZİ**

**MAKİNE ÖĞRENMESİ YÖNTEMLERİ İLE  
İNSAN AKTİVİTE TESPİT SİSTEMİ TASARIMI**

**KAZIM KIVANÇ EREN**

**KOCAELİ 2019**

KOCAELİ ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
BİLGİSAYAR MÜHENDİSLİĞİ  
ANABİLİM DALI

YÜKSEK LİSANS TEZİ

MAKİNE ÖĞRENMESİ YÖNTEMLERİ İLE  
İNSAN AKTİVİTE TESPİT SİSTEMİ TASARIMI

KAZIM KIVANÇ EREN

Doç. Dr. Kerem KÜÇÜK  
Danışman, Kocaeli Üniversitesi  
Dr. Öğr. Üyesi Orhan AKBULUT  
Jüri Üyesi, Kocaeli Üniversitesi  
Doç. Dr. Cüneyt BAYILMIŞ  
Jüri Üyesi, Sakarya Üniversitesi



Handwritten signatures of the supervisors and jury members, including Kerem Küçük, Orhan Akbulut, and Cüneyt Bayılmış.

Tezin Savunulduğu Tarih: 22.01.2019

## ÖNSÖZ VE TEŞEKKÜR

Teknolojinin çok hızlı bir şekilde gelişmesi ile mobil cihazlar günlük yaşantının sıradan bir unsuru haline gelmiştir. Bu gelişmenin mobil cihazlardaki sensörlere de yansımalarının bir sonucu olarak, mobil platformda çalışmak üzere geliştirilebilecek uygulamaların çeşitliliği de artmaktadır. Bu alanlardan biride sağlık hizmeti alanıdır. Bu alanda kişilerin hareketlerinin analiz edilmesi, yaşlılar için güvenli takip uygulamaları, kilo kontrolü gibi çeşitli alanlarda önem taşıyan bir uygulama haline gelmiştir. Bu açıdan kişi hareketlerinin daha iyi bir biçimde tespit edilmesi bu alanın gelişmesi için büyük önem taşımaktadır.

Bu tez çalışmasında, makine öğrenmesi yöntemleri kullanılarak insan aktivite tespit sistemi tasarımı üzerinde çalışılmıştır. Farklı özniteliklerin araştırılarak insan aktivitelerinde kullanılan özniteliklerin bu sistemlerdeki önemi incelenerek, insan aktivite tespit sistemlerinde kullanılacak özniteliklerin daha iyi anlaşılması amaçlanmıştır. Ayrıca var olan problemler incelenmiş ve problemlerin çözümü hususunda neler yapılabileceği irdelenmiştir.

Tez çalışması boyunca büyük bir sabır ve özveri ile bilgi birikimi ve deneyimlerini paylaşan, her zaman şahsım için motivasyon kaynağı ve iyi bir örnek olan danışmanım Sayın Doç. Dr. Kerem KÜÇÜK'e saygılarımı ve teşekkürlerimi bir borç bilirim.

Her zaman ve her konuda yanımda olan, eğitimin insan hayatındaki en önemli unsurlardan olduğunu öğreten saygıdeğer babam Eşref EREN ve sevgili annem Selma EREN'e, çalışmalarım esnasında her zaman pozitif ve motive edici yönüyle yanımda olan sevgili ablam Gizem EREN UYSAL'a sevgi ve teşekkürlerimi bir borç bilirim.

Ocak 2019

Kazım Kıvanç EREN

## İÇİNDEKİLER

ÖNSÖZ VE TEŞEKKÜR .....	i
İÇİNDEKİLER .....	ii
ŞEKİLLER DİZİNİ .....	iii
TABLolar DİZİNİ.....	iv
SİMGELER VE KISALTMALAR DİZİNİ .....	v
ÖZET .....	vi
ABSTRACT .....	vii
GİRİŞ.....	1
1. KULLANILAN YÖNTEMLER .....	4
1.1. Makine Öğrenmesi .....	4
1.1.1. Sınıflandırma problemi .....	5
1.1.2. K-en yakın komşu algoritması .....	6
1.1.3. Karar ağaçları .....	7
1.1.4. Rastgele orman algoritması (random forest - RF).....	8
1.2. Örüntü Tanıma.....	9
1.2.1. Markov zincirleri (Discrete Markov chains) .....	10
1.2.2. Saklı Markov modelleri .....	11
1.2.3. Olabilirlik hesabı.....	13
1.3. İnsan Aktivite Tespiti ile İlgili Çalışmalar.....	16
1.3.1. Sınıflandırma yöntemleri ile insan aktivitelerinin tanımlanması .....	17
1.3.2. Saklı Markov modelleri ile insan aktivitelerinin tanımlanması.....	18
2. İNSAN AKTİVİTE TESPİTİ İÇİN SİSTEM MİMARİSİ.....	20
2.1. İnsan Aktivitesi Tanımlama Aşamaları .....	20
2.1.1. Veri toplama.....	20
2.2. Öznitelik Çıkarımı.....	23
2.2.1. Sürekli zamanlı dalga özniteliklerinin kullanılması .....	23
2.2.2. Olasılıksal özniteliklerin kullanılması .....	25
2.3. Sistem Dizaynı .....	27
3. DENEYSEL ÇALIŞMALAR .....	29
3.1. Veri Setlerinin Hazırlanması.....	29
3.1.1. Ham veri seti .....	29
3.1.2. Sürekli zamanlı dalga öznitelikleri .....	30
3.1.3. HMM (olasılıksal) öznitelikleri .....	30
3.2. Gerçekleştirilen Deneyler.....	35
3.2.1. Sürekli zamanlı öznitelikler üzerinde gerçekleştirilen deneyler .....	36
3.2.2. HMM öznitelikleri üzerinde gerçekleştirilen deneyler .....	38
4. GERÇEK ZAMANLI UYGULAMA ÖRNEĞİ.....	41
5. SONUÇLAR VE ÖNERİLER .....	43
KAYNAKLAR .....	47
KİŞİSEL YAYIN VE ESERLER .....	51
ÖZGEÇMİŞ .....	52

## ŞEKİLLER DİZİNİ

Şekil 1.1. k-En Yakın Komşu algoritmasına ait kaba kod .....	6
Şekil 1.2. Rastgele Orman algoritması çalışma şeması .....	9
Şekil 1.3. Örnek bir Saklı Markov Modeli .....	12
Şekil 1.4. Dünya genelinde mobil cihaz kullanım sayısı.....	16
Şekil 2.1. 3-eksenli ivmeölçere sahip bir mobil telefon .....	21
Şekil 2.2. Gerçek zamanlı hareketlere ait ivme değerleri.....	23
Şekil 2.3. İnsan aktivite tespit sistemi için gerçekleştirilen aşamalar.....	28
Şekil 3.1. Geçiş olasılıklarının hesaplanması .....	32
Şekil 3.2. Başlangıç olasılıklarının hesaplanması.....	33
Şekil 3.3. İnsan aktivite tespit sistemi için gerçekleştirilen aşamalar.....	35
Şekil 4.1. İstemci – Sunucu Yapısı .....	41
Şekil 5.1. Bir mobil telefonun alabileceği farklı şekiller .....	45

## TABLolar DİZİNİ

Tablo 1.1. Bir Markov Zincirinin Bileşenleri .....	11
Tablo 1.2. Saklı Markov Modelinin bileşenleri .....	12
Tablo 1.3. Saklı Markov Modellerdeki 3 Temel Problem .....	13
Tablo 2.1. Olasılıksal öznitelikler kullanan veri seti .....	27
Tablo 3.1. Veri setindeki insan aktivitelerinin dağılımı .....	29
Tablo 3.2. Veri setindeki kayıtlara ait öznitelikler .....	29
Tablo 3.3. Saklı Markov modeli durumları .....	32
Tablo 3.4. Elde edilen öznitelikler .....	34
Tablo 3.5. İnsan aktiviteleri için kısaltma tablosu.....	36
Tablo 3.6. C4.5 Karar Ağacı.....	36
Tablo 3.7. Rastgele Ağaçlar (Random Forest – RF) .....	36
Tablo 3.8. SVM-Çoklu Çekirdek (Poly Kernel) .....	37
Tablo 3.9. SVM-RBF Çekirdeği.....	37
Tablo 3.10. Çok Katmanlı Algılayıcı (Multilayer Perceptron – MLP) .....	37
Tablo 3.11. K-NN (1-NN) .....	37
Tablo 3.12. C4.5 Karar Ağacı.....	38
Tablo 3.13. Rastgele Ağaçlar.....	38
Tablo 3.14. SVM- Çoklu Çekirdek.....	39
Tablo 3.15. SVM-RBF Çekirdeği.....	39
Tablo 3.16. MLP .....	39
Tablo 3.17. 1-NN .....	39
Tablo 4.1. Gerçek Zamanlı Uygulama için Katışıklık Matrisi.....	42

## SİMGELER VE KISALTMALAR DİZİNİ

$\sigma$	: Standart Sapma
$\lambda$	: Saklı Markov Modeli
D	: Veri Seti

### Kısaltmalar

API	: Application Interface (Uygulama Arayüzü)
DP	: Decision Profile (Karar Profilleri)
DTs	: Decision Templates (Karar Şablonları)
DT	: Decision Tree (Karar Ağacı)
HAR	: Human Activity Recognition (İnsan Aktivitelerinin Tanımlanması)
HMM	: Hidden Markov Models (Saklı Markov Modelleri)
HMME	: Hidden Markov Model Ensemble (Saklı Markov Modellerin Birleşimi)
Hz	: Hertz
ID3	: Iterative Dichotomiser (İteratif İkiye Ayırıcı)
k-NN	: K-Nearest Neighbors (K-En Yakın Komşu Algoritması)
MAD	: Mean Absolute Deviation (Standart Ortalama Sapma)
MLP	: Multilayer Perceptron (Çok Katmanlı Algılayıcı)
NB	: Naive Bayes Classifier (Naif Bayes Sınıflandırıcı)
RF	: Rastgele Orman (Random Forest – RF)
RBF	: Radial Basis Function (Radyal Temelli Fonksiyon)
SVM	: Support Vector Machine (Destek Vektör Makineleri)
WEKA	: Waikato Environment for Knowledge Analysis (Bilgi Analizi için Waikato Ortamı)
WISDM	: Wireless Sensor Data Mining (Kablosuz Sensör Veri Madenciliği)

## MAKİNE ÖĞRENMESİ YÖNTEMLERİ İLE İNSAN AKTİVİTE TESPİT SİSTEMİ TASARIMI

### ÖZET

İnsan aktivitelerinin tespiti, mobil cihazların ve buna bağlı olarak içerisindeki sensörlerin gelişimi ve makine öğrenmesi yöntemlerinin giderek yaygınlaşması ile yaygın olarak çalışılan oldukça popüler bir problem haline gelmiştir. Bu problemin çözümü için, toplanan farklı hareketlere ait sensör verileri gerçekleştirilen hareketler ile etiketlenerek bir sınıflandırma problemine dönüştürülür. Toplanan veri üzerinde sınıflandırma algoritmaları gerçekleştirilerek farklı insan aktiviteleri ayırt edilmeye çalışılır.

İnsan aktivitelerinin tespiti, kişiye özel günlük kalorilerin hesaplanması, kişinin gerçekleştirdiği hareketlere göre sağlık durumunun analizi gibi sağlık uygulamaları olarak veya yaşlı insanların buldukları ortamda gerçekleştirdiği hareketler ile gözetilmesi, insan pozisyon takibi ve çeşitli güvenlik uygulamalarında kullanılmaktadır.

Bu tez çalışması kapsamında, bir insan aktivite tespit sistemi gerçekleştirilmiştir. WISDM (Wireless Sensor Data Mining) ham veri seti kullanılarak, sürekli zamanlı dalga öznitelikleri ve Saklı Markov Modellerine dayalı öznitelikler çıkarılmış ve bu özniteliklerin insan aktivite tanımlamadaki etkileri incelenmiştir. Kurulmuş olan makine öğrenmeleri algoritmaları kullanılarak gerçek zamanlı bir uygulama örneğinde bu tez çalışması kapsamında gerçekleştirilmiştir. Deneysel sonuçlar ile gerçek zamanlı uygulama arasındaki farklılıklar incelenmiş ve buna bağlı olarak insan aktivitelerinin tespitinde karşılaşılabilecek sorunlara değinilmiş ve bu sorunların çözümü hakkında öneriler sunulmuştur.

**Anahtar Kelimeler:** İnsan Aktivitelerinin Tanımlanması, Makine Öğrenmesi, Mobil Sensörler, Saklı Markov Modeli, Sınıflandırma.



## **A HUMAN ACTIVITY RECOGNITION SYSTEM BASED MACHINE LEARNING METHODS**

### **ABSTRACT**

Consequently the evolution of mobile sensors and the increasingly widespread use of machine learning methods, detection of human activities has become a prevalent problem. For the solution of this problem, the sensor data of the different activities collected are converted into a classification problem by being labeled with the movements performed. By using classification algorithms on the collected data, different human activities are tried to predict.

Human Activity Recognition (HAR) is used in health care applications such as the calculation of personal calories daily, analysis of health status of the person or observation of movements of elderly people for their surveillance, human position tracking or various security applications.

Within the scope of this thesis, a human activity detection system has developed. Using WISDM (Wireless Sensor Data Mining) raw data set, continuous time wave attributes and Hidden Markov Models based attributes were extracted and their effects on human activity detection were examined. We compared experimental results and real-time system, and consequently, the problems that may be encountered in defining human activities are discussed and the solution of these problems have proposed.

**Keywords:** Human Activity Detection, Machine Learning, Mobile Sensors, Hidden Markov Models, Classification.

## GİRİŞ

Günümüzde teknolojinin büyük bir hızla gelişmesi ve bu gelişmenin bir sonucu olarak mobil cihazlar gündelik yaşantının ayrılmaz bir parçası haline gelmiştir. Bu gelişmenin en çarpıcı örneğini mobil telefonlar oluşturmaktadır. Alman istatistik şirketi Statistica'nın yaptığı çalışma bu durumu destekler niteliktedir. Yapılan çalışmada dünya genelinde ortalama mobil telefon sayısının 2018 itibari ile yaklaşık olarak 2,53 milyar olduğu belirtilmiştir. Bu sayının giderek artarak 2020 yılı itibari ile toplam 2,87 mobil telefon sayısına ulaşılması beklenmektedir.

Mobil telefonlar dışında benzer bir durum giyilebilir cihazlarda da görülmektedir. Görsel Ağ Endeksi Global Mobil Veri Trafik Tahmini Raporu'na göre 2019 yılında dünyadaki giyilebilir cihazlar 109 milyondan 578 milyona ulaşacak ve mobil veri trafiğinin %97'si mobil cihazlar yoluyla üretilecektir [2].

Teknolojinin bu denli hızla gelişmesi ve mobil cihazların büyük oranda yaygınlaşması, doğal olarak bu cihazların donanımsal olarak bu gelişmeye ayak uydurması anlamına gelmektedir. Akıllı telefonlar her geçen gün daha iyi, hassas ve çeşitli donanımlarla donatılmaktadır. Günümüzde ortalama bir mobil telefonda GPS, yüksek çözünürlüklü kameralar farklı amaçlara hizmet eden hassas sensörler (ivmeölçer, jiroskop, ışık sensörleri gibi) artık standart hale gelmiştir. Mobil cihazların fiyatlarının giderek azalması, donanımsal açıdan verimliliklerinin artması, taşınabilirlik avantajları bahsi geçen gelişmelerde etkili olmuştur [3].

Donanımsal gelişmeler sayesinde akıllı telefonlardaki uygulamalar daha farklı ve spesifik problemlerin çözümü için kullanılabilir duruma gelmiştir. Bu çalışmalardan biri akıllı telefonlar kullanılarak insan aktivitelerinin tespiti. İnsan aktivitelerinin tanımlanması çok çeşitli alanlarda kullanılan ve uygulanması halen zor olan bir çalışma alanıdır. Sağlık alanında kişilerin günlük hareketlerinin analizinin yapılması, yaşlı insanlar için düşme tespiti uygulamaları, akıllı evler, ev güvenliğinin sağlanması gibi çok çeşitli alanlarda kullanılmaktadır [4-8].

İnsan aktivitesi tanımlanması ile alakalı temelde iki yaklaşım bulunmaktadır. Bu yaklaşımlardan birincisi, bir ortama farklı konumlarda yerleştirilmiş çeşitli kameralar aracılığıyla, ortamda bulunan kişi veya kişilerin hareketlerinin görüntü işleme

yöntemleri ile tespit edilmesidir [9-12]. Fakat bu yöntemler kameraların kısıtlı bir alana yerleştirilmesine, dolayısıyla geliştirilecek bir uygulamanın kısıtlı mekânda geliştirilebilmesine sebep olmaktadır. Görüntü işleme dayalı yöntemler geniş alanlarda kullanılsa da bir altyapı desteğinin varlığına ihtiyaç duymaktadır. Ayrıca sistem kurulum maliyetleri bahsedilecek olan veri madenciliği yöntemlerine oranla daha fazladır.

İnsan aktivitesi alanında kullanılan ve giderek yaygınlaşan bir diğer yöntem ise ilgili problemin çözümünde veri madenciliği yöntemlerinin kullanılmasıdır. Bu yöntem genel olarak farklı hareket çeşitleri gerçekleştirilirken vücudun farklı kısımlarına yerleştirilmiş sensörlerden elde edilen verilerin veri madenciliği metotlarına uygun olarak işlenerek insan aktivitelerinin farklı sınıflar olarak görülüp ilgili problemin bir sınıflandırma problemine dönüştürülerek bu problemin çözülmesi esasına dayanmaktadır. Mobil sensörlerden elde edilen ham veri üzerinden oluşturulan veri seti üzerinden farklı sınıflandırma algoritmaları kullanılarak farklı hareket türlerinin tespiti yapılmaya çalışılmaktadır. Karar Ağaçları (Decision Tree - DT), Naif Bayes Sınıflandırıcı (NB), Çok Katmanlı Algılayıcılar (Multilayer Perceptron - MLP), Destek Vektör Makineleri (Support Vector Machines - SVM) gibi farklı algoritmalar bu türden çalışmalarda sıklıkla kullanılmaktadır.

Bu tez çalışmasında, bir mobil cihaz için 3-eksenli ivmeölçer verilerini kullanarak bir insan aktivite tespit sistemi tasarımı yapılmıştır. İnsan aktivitelerindeki sürekli zamanlı dalga öznitelikleri ve Saklı Markov Modellerine dayalı iki farklı öznitelik uzayında çalışılmıştır. Bu iki öznitelik uzayı kullanılarak farklı makine öğrenme algoritmaları ile sınıflandırma modelleri kurularak, hem insan aktivitelerindeki başarılı modeller incelenmiş hem de bir insan aktivite tespit sistemi tasarımında kullanılacak farklı öznitelik uzayları incelenmiştir. Gerçek zamanlı bir aktivite tespit sistemindeki olası zorluklar ve bu olası zorluklara çözüm önerileri sunulmuştur.

Tezin çalışmasının organizasyonu şu şekildedir: Birinci bölümde, tez çalışması kapsamında kullanılan yöntemler ve insan aktivite tespiti ile ilgili yapılmış literatür çalışmalarından bahsedilmiştir. İkinci bölümde insan aktivite sistemi için gerçekleştirilen sistem tasarımından bahsedilmiştir. Üçüncü bölümde bu sistemin gerçekleştirilmesi için kullanılan farklı veri setinin oluşturulması ile alakalı adımlar ve bu veri setleri üzerinde gerçekleştirilen deneyler anlatılmıştır. Farklı veri setleri kullanılarak oluşturulan insan aktivitesi modelleri incelenmiş ve başarımları karşılaştırılmıştır. Dördüncü bölümde gerçek zamanlı bir senaryo üzerinde kurulan

bir insan aktivite tespit sistemine ait sonuçlar incelenmiştir. Beşinci ve son bölümde, deneysel sonuçlar özetlenmiş, gerçek uygulama sonuçları ile karşılaştırılmış ve gerçek bir uygulamada yapılması gerekenler ve insan aktiviteleri ile alakalı teknik zorluklardan bahsedilerek bu teknik zorluklara ilişkin olası çözümler sunulmuştur. Ayrıca gerçekleştirilmesi planlanan gelecek çalışmalardan da bahsedilmiştir.



## 1. KULLANILAN YÖNTEMLER

Bu çalışma kapsamında insan aktivitelerinin tespiti için, makine öğrenmesi ve örüntü tanıma dayalı yaklaşımlar kullanılmıştır. Bölüm 2’de açıklanan “Sistem Mimarisi” esasında bu iki kavram üzerine kurulmuştur. Bu bölümde, makine öğrenmesi ve örüntü tanıma problemlerine ilişkin teknik detaylar verilmiştir. Bu teknik detaylar geniş çaplı bir şekilde sunulmaktan ziyade, bu tez çalışması kapsamında kullanılan yöntemler ile sınırlı kalınarak sunulmuştur. İlgili teknik detaylar ve bu çalışmada kullanılan yöntemler sunulduktan sonra, konu ile ilgili literatürde bahsi geçen teorik konularla alakalı olan çalışmalar sunulmuştur.

### 1.1. Makine Öğrenmesi

Makineler düşünebilir mi? Alan M. Turing, bu türden bir soruyu sormuş ve kendi perspektifinden bu soruyu yorumlayarak, “öğrenen makineler” olarak adlandırılacak ve yapar bir zekâyâ sahip olabilecek makineler fikrini sunmuştur [12]. Modern dönemde makine öğrenmesi olarak adlandırılan bu kavram, bir bilgisayarın çözmesi beklenen bir spesifik problemin, çeşitli algoritmalar ve istatistiksel metotların kullanımı ile çözüm performansının artırılması olarak adlandırılabilir.

Turing bu fikri 1950 yılında sunmuş olsa da, makine öğrenmesi kavramının temelleri çok daha eskiye dayanmaktadır. 1805 yılında Legendre, günümüzde “En Küçük Kareler Yöntemi” (Least Squares Method) olarak adlandırılan “Eğri Uydurma Metodu” (Curve Fitting) sunmuştur [13]. 1812’de Laplace, 1763 yılındaki Thomas Bayes’in ana çalışmasını temel alarak bugün Bayes Teoremi olarak bildiğimiz koşullu olasılığa dayalı olabilirlik teoremini tanımlamıştır [14, 15]. Markov, kendi adıyla anılan ve ilk kez bir şiirin analizinde kullandığı Markov Zincirlerini (Markov Chains) tanımlamıştır (bu tez çalışmasında Markov Zincirlerini temel alan Saklı Markov Modelleri (Hidden Markov Models – HMM) kullanılmıştır [16].

Teknolojinin evriminin bir sonucu olarak, problemlerin çeşitliliği ve karmaşıklığı artmış, bu problemlerin çözümleri için standart bilgisayar programlar tarafından çözülmesi imkânsız hale gelmiştir. Bir makine öğrenmesi süreci, var olan veriyi ve geçmiş deneyimleri kullanarak kendi performanslarını iyileştirecek bilgisayar

programlarının oluşturulmasına imkân vererek bu problemler için hedef bir çözüm oluşturmaktadır.

Makine öğrenmesi, bir veri yığınının bu verinin içerisinde barındırdığı bilginin çıkarılması ve öğrenilmesi esasına dayanmaktadır. Bir veri seti içerisinde belirli gizli örüntüler (patterns) olduğu varsayımı yapıldığında, bir makine öğrenme süreci veri seti içindeki benzerlik ve farklılıkları kullanarak, veri seti içerisindeki bu örüntülerin açığa çıkarılması ve uygulanma amacına yönelik olarak kullanılmasıdır.

Bu bölümün devamında, gerçekleştirilen tez çalışması kapsamında kullanılan farklı makine öğrenme algoritmaları –sınıflandırma problemi için- hakkında bilgiler verilmiştir.

### 1.1.1. Sınıflandırma problemi

Sınıflandırma, makine öğrenmesi problemlerinden biri olup, en sık karşılaşılan kavramlardandır. Farklı türden verilerin aynı özellikler çerçevesinden değerlendirildiği bir durum varsayılmış olsun. Bu duruma örnek olarak, kilo ve boy özelliklerine bakarak kişilerin cinsiyetlerinin tespit edildiği bir senaryo hayal edilebilir. Burada tespit edilmesi beklenen, hedeflenen durum kişinin cinsiyet bilgisidir. Bu problemde cinsiyet sınıf olarak tanımlanmaktadır. Bu sınıfın tespiti için gerekli olan (bu senaryoda boy ve kilo) özelliklere ise öznitelik adı verilmektedir.

Daha biçimsel olarak ifade edilmek istenirse  $D$ , bir  $N$  boyutlu veri seti olsun:

$$D = \{(X^1, Y^1), (X^2, Y^2), (X^3, Y^3), \dots, (X^N, Y^N)\} \quad (1.1)$$

Burada  $X$  öznitelik uzayını,  $Y$  ise etiket bilgisidir. Bir sınıflandırma probleminde temel amaç  $X$  öznitelik uzayından faydalanılarak, bu öznitelik uzayı ile  $Y$  etiket bilgisi arasındaki saklı bir ilişkinin tespit edilmesidir. Bu şekilde bir ilişki ortaya konduğunda, daha önce hiç karşılaşılmayan ve etiketi ( $Y$  değeri) bilinmeyen  $X$  öznitelik uzayından gelmiş bir veri noktasının etiketinin tespiti yapılmaya çalışılmaktadır.  $X$  öznitelik uzayının kullanılarak  $Y$  etiketinin tespit edilmeye çalışılması makine öğrenmesi alanında, “Sınıflandırma Problemi” olarak adlandırılmaktadır.

Sınıflandırma probleminde etiketli bir veri seti elimizde bulunmaktadır. Bu türden etiketli verilerin sınıflandırma probleminde kullanıldığı durum, “Denetimli Öğrenme” (Supervised Learning) olarak adlandırılmaktadır. Herhangi bir etiket bilgisinin bulunmadığı durumlarda ise veride bulunan ilişkilerin çıkarılması için “Denetimsiz

Öğrenme” (Unsupervised Learning) adı verilen farklı bir öğrenme metodolojisi de bulunmaktadır. Bu tez çalışması kapsamında denetimli öğrenme problemi üzerinde durulmuştur.

### 1.1.2. K-en yakın komşu algoritması

K-en yakın komşu algoritması (k-nearest neighbors, k-NN), örnek tabanlı öğrenmenin (instance-based learning) bir alt kategorisi olarak ifade edilebilir. Örnek tabanlı yöntemler esasında, hiç bir öğrenme mekanizması gerçekleştirilmemektedir. Gelen bir örneğin, diğer örneklerle benzerliğine anlık bakarak karar vermektedir. Dolayısıyla bu türden yöntemler aynı zamanda tembel öğrenme (lazy learning) olarak adlandırılmaktadır.

K-en yakın komşu algoritmasına ait kaba kod (psuedo-code) Şekil 1.1’de verilmiştir.

```
1 k_NN(X[N], Y[N], x)
2   //X:Eğitim Seti, Y: X'e ait etiketler, x: Sınıfı tespit edilecek örnek
3   FOR i = 1 TO M
4     d(X[i], x) mesafelerini hesapla
5   END FOR
6   en_kucuk_mesafeli_k_tane_noktayı_bul
7 RETURN çoğunluk oylamasıyla sınıfı belirle
```

Şekil 1.1. k-En Yakın Komşu algoritmasına ait kaba kod

Burada  $d(X[i], x)$  ile belirtilen metrik uzaklık metriğidir. Kaba koddan da anlaşılacağı üzere gelen bir veri noktası için, veri setindeki diğer tüm noktalara olan uzaklık hesabı yapılır. Elde edilen bu uzaklık değerleri içerisinde en yakın k komşunun sınıflarına bakılır. K-En yakın komşular arasından en çok hangi sınıfa ait örnek varsa o sınıf yeni gelen verinin sınıfı olarak belirlenmektedir. Uzaklık metriği olarak Öklid uzaklığı, Manhattan uzaklığı ve Minkowski uzaklığı sıklıkla kullanılmaktadır.

k-NN gibi, örnek tabanlı bir öğrenme sistemi aşağıdaki olgulara gereksinim duymaktadır [17]:

- Veri noktaları (örnekler) arasındaki benzerliği tanımlayacak bir uzaklık fonksiyonu
- k ile belirtilen, bir veri noktası için gerçekleştirilecek bir operasyon üzerinde kaç farklı komşu veri noktasına bakılacağı bilgisi
- Sistem başarısını ve öğrenme kalitesini arttırmak için k-komşu arasında bir ağırlıklandırma metodolojisi

- Verilen problemin çözümü için, komşuların kullanımını belirleyen bir değerlendirme yöntemi

### 1.1.3. Karar ağaçları

Karar ağaçları, parametrik olmayan ve sınıflandırma algoritmalarında kullanılan bir denetimli öğrenme algoritmasıdır. Çok farklı karar ağacı yapıları olmakla beraber, bu çalışmada da kullanılan ve Ross Quinlan tarafından sunulan C4.5 algoritmasıdır. C4.5 entropi tabanlı bir karar ağacı algoritmasıdır [18]. Daha önce yine Quinlan tarafından 1986 da sunulan ID3 (Iterative Dichotomiser) algoritmasının bir eklentisi olarak düşünülebilir [19].

C4.5 ile ID3 arasındaki en önemli iki fark, C4.5'in hem kategorik hem de sürekli zamanlı öznitelige sahip veriler (iki öznitelik türünün kombinasyonu da olabilir) üzerinde çalışabilir özellikte olmasıdır. ID3 ise yalnızca kategorik öznitelik uzayına sahip veriler üzerinde uygulanmaktadır. Ayrıca budama denilen (pruning) aşaması da C4.5'te bulunurken, ID3'te bu aşama bulunmamaktadır. Dolayısıyla C4.5 aşırı öğrenme (overfitting) problemine karşı daha dirençlidir.

Bir ağaç veri yapısı kök düğümden başlayarak aşağı doğru dallanan bir yapıdadır. Bir karar ağacında temel amaç benzer özellikler gösteren veri noktalarını aynı bölgede toplamaktır. Yani ağaçtaki bir kök düğüme ulaşıldığında olabildiğince homojen bir durumun olması istenmektedir. Bu homojenliğin sağlanması için entropi kullanılmaktadır.

C4.5 algoritması için, bir veri setindeki toplam iki farklı entropi değerinin hesaplanması gerekmektedir. İlk aşama olarak C farklı sınıf sayısı olmak üzere veri setindeki toplam entropi değeri hesaplanmaktadır. Bu değer Eşitlik 1.2'de gösterilmiştir.

$$\text{Bilgi}(D) = - \sum_{i=1}^C p_i \log_2 p_i \quad (1.2)$$

Burada  $p_i$ , i sınıfının tüm veri seti içerisinde gelme olasılığıdır. Bir karar ağacının ayrıldığı düğümlere karar düğümleri denmektedir. Bu karar düğümleri esasında ağacı (dolayısıyla verileri) öznitelik uzayındaki özniteliklerin değerlerine göre ayırmaktadır.



Bir karar ağacında her bir karar düğümüne gelindiğinde iki alt ağaç kendi içlerinde benzer veri noktalarını temsil etmektedir. Ağacın olabildiğince dengeli olması için karar düğümlerinde temsil edilen özniteliklerin verileri ayırmada en başarılı olan özniteliklerden daha az başarılı olanlara doğru gitmesi gerekmektedir. Bu sebeple ikinci aşama olarak her bir öznitelik için entropi değeri hesaplanmaktadır. Bu değer bir özniteliğin diğer özniteliklere üstünlük oranını ifade etmektedir. Kesikli X özniteliği için entropi değeri, K, X özniteliğinin alabileceği değerler olmak üzere, D veri seti için:

$$\text{Bilgi}_X(D) = \sum_{i=1}^K p_i \text{Bilgi}(S_i) \quad (1.3)$$

şeklindedir.

1.2 ve 1.3'e bağlı olarak X özniteliği için bilgi kazancı değeri Eşitlik 1.4'te gösterilmiştir.

$$\text{Kazanç}(X) = \text{Bilgi}(D) - \text{Bilgi}_X(D) \quad (1.4)$$

Bilgi kazancı en yüksek olan öznitelik alınarak ağacının karar düğümü olarak eklenir. Bu düğümün dalları iki alt ağaç oluşturmaktadır. İki alt ağaç kendi içlerinde, alt ağaçlar arasından daha yüksek bir benzerlik göstermektedir. Bir karar düğümü belirlendikten sonra kalan öznitelikler, öz yinelemeli olarak, herhangi bir öznitelik kalmayıncaya kadar bilgi kazancına göre sıralanarak en büyüğü ağacın ayrılma noktası olacak şekilde ağaca eklenmektedir. Ağaçlar kullandıkları budama aşamalarından dolayı aşırı öğrenmeye karşı güçlü modeller oluştururlar.

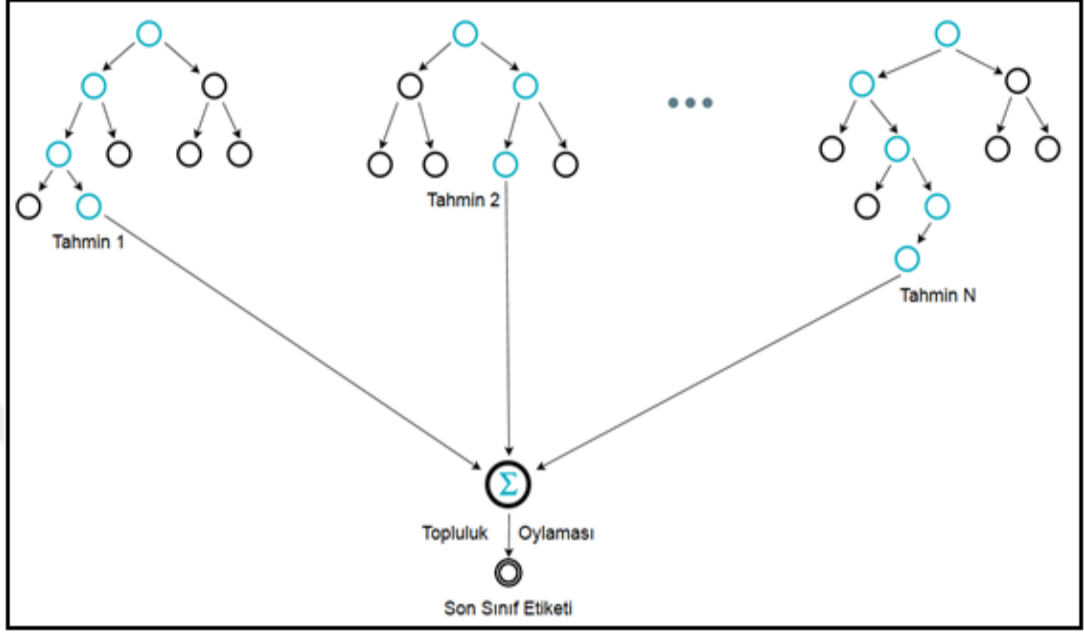
#### 1.1.4. Rastgele orman algoritması (random forest - RF)

Rastgele orman algoritması da karar ağaçları gibi parametrik olmayan, denetimli öğrenmede kullanılan bir makine öğrenmesi algoritmasıdır. Temel olarak, bir rastgele orman algoritması, veri setini kullanarak birbirine paralel farklı ağaçlar oluşturmaktadır. Sınıflandırılacak olan veri seti bu farklı ağaçlara giriş olarak sokularak her bir ağaç için birer sonuç elde edilir. Daha sonra bu sonuçlar birleştirilerek çoğunlukçu oylama (majority voting) işlemi uygulanır. Ağaçların çoğunluğu veri örneği için hangi sınıfı uygun görmüşse o değer sınıflandırma sonucu olarak seçilmektedir.

Topluluk oylaması (majority voting) ile Genelleme hatası (generalization error) ve aşırı öğrenme kusurları olabildiğince azaltılmaktadır.

Bu sayede veri seti için güçlü bir öğrenme modeli oluşturulmuş olunur.

Bu süreç Şekil 1.2'de gösterilmiştir.



Şekil 1.2. Rastgele Orman algoritması çalışma şeması

Özetlenecek olursa Rastgele Orman Algoritması 4 temel aşamadan oluşmaktadır [20]:

- k adet örnek alt veri seti eğitim setinden rastgele olarak seçilir
- k tane farklı Karar Ağacı oluşturulur ve her bir düğüm için
- k kez d tane öznelik değiştirilemeden seçilerek, karar ağaçlarındaki bilgi kazanımı yöntemine benzer bir öznelik ayırıcı bir yöntemle karar düğümleri tespit edilir.
- Elde edilen ağaçlar ile sınıflandırmalar yapılarak çoğunluk oylamasına tabi tutulur. En fazla sayıda seçilen sınıf nihai sınıf olarak seçilmektedir.

Rastgele ormanlar çoğunlukçu oylama kullandıklarından ötürü güçlü öğrenme modelleri oluşturmaktadırlar.

## 1.2. Örüntü Tanıma

Günümüzde yaygın olarak çözümü ile uğraşılan problemlerden biride, birbirine bağlı bir dizi olaydan oluşan gözlem serilerinin tespiti veya tanımlanması olarak adlandırabileceğimiz, örüntü tanıma problemidir. Bir örüntü doğası gereği iki gözlem noktası arasında belirli bir kural çerçevesi içerisinde bağımlılık arz eden bir yapı

taşımaktadır. Çevremizde bir örüntü olarak bahsedebileceğimiz çok fazla sayıda olgu bulunmaktadır.

Tipik bir örüntü örneği olarak dünyadaki hava durumu gösterilebilir. Bir hava durumunun yapısı temel olarak çok fazla parametreye bağlıdır: Dünyanın eksen eğikliği, Güneş'e olan mesafesi, dünya atmosferindeki gazların yapısı gibi. Bu parametrelerin doğa yasaları gereğince oluşturduğu sistemin bir sonucu olarak bizler mevsimler olarak adlandırdığımız hava koşulları değişimlerini gözlemliyoruz. Temel olarak bu parametreler değişmediği sürece – en azından periyodik olarak düzenli bir biçimde değiştiği varsayıldığında- bizler bu parametrelerin dönemlere göre farklılıkları arasındaki ilişkilere bakarak hava durum değişimlerini belirli olasılıklar dâhilinde tespit edebiliyoruz. Bilgisayar bilimleri açısından bakıldığında mobil cihazlardaki klavyelerde bulunan otomatik kelime tahmin ediciler gösterilebilir. Bu problem esasında bir dilin kelimeleri arasında olasılıksal bir dağılımın olduğu kabulüne göre hareket etmektedir. Türkçe açısından incelendiğinde, bir cümle için, bir kelimedenden sonra gelecek sıradaki kelimenin belirli bir olasılık dağılımı olduğu söylenebilir. Örneğin, “Merhaba” kelimesinden sonra “nasılsın” kelimesinin gelmesi, “matematik” kelimesinin gelmesinden daha olası olacağı sezgisel olarak dahi tahmin edilebilir bir şeydir. Kelime önerici uygulamalar da esasında bir kelime veya kelime grubundan (birden fazla kelimenin yan yana gelmesi) sonra gelecek kelimenin ne olacağını tahmin edilmesi ve bunu kişiye göstermesi temelinde çalışmaktadır.

İşte kendinden önceki gerçekleşmiş bir dizi olguya belirli parametrelerce olasılıksal olarak “bağımlı” olan, rassal süreç tarafından üretilen durumların nasıl modellendiği ile alakalı tartışılmaktadır [21].

### 1.2.1. Kesikli Markov zincirleri (Discrete Markov chains)

Bir Markov zinciri, bir t anında gerçekleşen bir durumun kendinden önce gelen bir dizi durumun zincirleme gerçekleşmesinin bir sonucu olduğuna, olasılıksal bir bağımlılık içerdiğini varsayan bir rasgele süreç olarak tanımlanabilir.

Daha biçimsel olarak gösterecek olursak,  $q_1, q_2, q_3, \dots, q_N$  durum değişkenleri olmak üzere, yukarıda bahsedilen durum Eşitlik 1.5'te gösterilmiştir.

$$P(q_t | q_1 \dots q_{t-1}) \quad (1.5)$$

Markov Varsayımı (Markov Assumption) olarak adlandırılan durumda, t zamanında gerçekleşen bir durum sadece t-1 anına bağlı olarak gösterilebilir:

$$P(q_t | q_{t-1}) \quad (1.6)$$

Bu tanıma ayrıca Birinci Dereceden Markov Modeli’de denmektedir. Temel amaç, işlem yükünü azaltmak olarak ifade edilebilir.

Bir Markov Zincirini tanımlayan bileşenler Tablo 1.1’de gösterilmiştir.

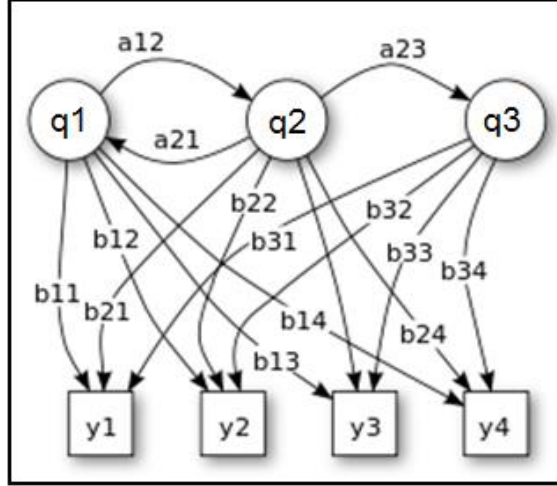
Tablo 1.1. Bir Markov Zincirinin Bileşenleri

$Q = q_1, q_2, q_3, \dots, q_N$	Markov Zinciri Durumları (States)
$A = a_{11}, a_{12}, a_{13}, \dots, a_{NN}$	Geçiş olasılıkları (transition probabilities), her bir $a_{ij}$ esasında i. durumdan j. duruma geçiş olasılığını gösterir. ( $\sum_{j=1}^N a_{ij} = 1$ )
$\pi = \pi_1, \pi_2, \pi_3, \dots, \pi_N$	Durum değişkenleri üzerindeki başlangıç olasılık değerlerinin dağılımı. Markov zincirinin i. Durumdan başlama olasılığıdır. ( $\sum_{j=1}^N \pi_{ij} = 1$ )

### 1.2.2. Saklı Markov modelleri

Doğrudan gözlemleyebildiğimiz durumların birleşik olasılıklarını hesap etmeye çalıştığımızda, Markov Zincirleri gayet kullanışlı olabilir. Fakat bazen bir olgu, gözlemleyebildiğimiz olaylara bağlı ve doğrudan gözlemleyemediğimiz durumların zincirleme olarak birlikte gerçekleşmesine bağlı olarak meydana gelebilir. Bu türden olayların gerçekleşme olasılıklarının hesaplanmasında gözlemlenen olaylar zinciri ve gözlemlenemeyen olaylar zinciri arasında bir ilişkiye dayalı birleşik olasılık modellerinin kurulması gerekmektedir. Gerçekleşen olaylar zincirinin belirli bir olasılığı olduğu gibi gözlemlenemeyen saklı durumların da gerçekleşen olaylara bağlı bir zincirleme olasılıkları bulunmaktadır.

Bu durum için verilebilecek en tipik örneklerden birisi kelimelerin gelme olasılıklarının sadece önceki kelimeye göre değil, kelimenin cümledeki hangi ögesi olduğuna bağlı olması durumudur. Örneğin ardışık iki kelime gelmiş olsun, bu kelimelerin cümle içerisindeki yerine göre isim veya fiil hatta sıfat oluşları arasındaki ilişki de sadece iki kelimenin ardışık olarak yan yana gelmesi bir önem taşıyabilir. Kelimelerin ardışık olarak gelmelerinin yanında, bu kelimelerin oluşturabileceği farklı kelimelerin birlikte geçtiklerinde hangi cümle ögesi olduklarının olasılıklarında hesaba katılmalıdır.



Şekil 1.3. Örnek bir Saklı Markov Modeli

Burada doğrudan gözlemleyebildiğimiz olgular kelimelerdir. Bu kelimelerin doğrudan gözlemleyemediğimiz saklı niteliklerinin de (kelimelerin birlikte geçtiği durumlarda yapılarının isim, sıfat, fiil olma olasılıkları gibi) durumların da var olduğu kabul edilerek modeller kurulabilir. Saklı durumların bulunduğu bu yapılar, Saklı Markov Modelleri (Hidden Markov Models – HMM) olarak adlandırılmaktadır. Bir saklı markov model örneği Şekil 1.3'te gösterilmiştir. Şekil 1.3'te belirtilen Saklı Markov modelinin bileşenleri ise Tablo 1.2'de gösterilmiştir:

Tablo 1.2. Saklı Markov Modelinin bileşenleri

$Q = q_1, q_2, q_3, \dots, q_N$	Markov Zinciri Durumları (States)
$A = a_{11}, a_{12}, a_{13}, \dots, a_{NN}$	Geçiş olasılıkları (transition probabilities), her bir $a_{ij}$ esasında i. durumdan j. duruma geçiş olasılığını gösterir. ( $\sum_{j=1}^N a_{ij} = 1$ )
$O = O_1, O_2, O_3, \dots, O_T$	Bir T boyutlu gözlem serisi. Doğrudan gözlemleyebildiğimiz olayları ifade etmektedir.
$B = b_i(O_t)$	Emisyon olasılığı (emission probability). i. durum için (state) ilgili gözlemin gelme olasılığı.
$\pi = \pi_1, \pi_2, \pi_3, \dots, \pi_N$	Durum değişkenleri üzerindeki başlangıç olasılık değerlerinin dağılımı. Markov zincirinin i. Durumdan başlama olasılığıdır. ( $\sum_{j=1}^N \pi_{ij} = 1$ )

Bir HMM, kısaca  $\lambda = (A, B, \pi)$  olarak gösterilebilir.  $\lambda$  ilgili HMM'i gösterir, A geçiş olasılıklarını, B emisyon olasılıklarını,  $\pi$  ise başlangıç olasılıklarını temsil eder.

Bir HMM'de geçiş olasılıkları Eşitlik 1.7'de gösterilmiştir.

$$a_{ij} = P(q_{t+1} = q_j | q_t = q_i) \quad (1.7)$$

İlgili HMM'e ait emisyon olasılıkları Eşitlik 1.8'de gösterilmiştir.

$$b_j(O_t) = P(O_t = O_j | q_t = q_j) \quad (1.8)$$

Burada bir gözlem değeri kesikli bir değer olmak zorunda olmayabilir. Dolayısıyla bir saklı durum için emisyon olasılığı sürekli zamanlı veriler için tekrar düzenlenebilir. Bu durumda emisyon olasılıkları Eşitlik 1.9'da gösterildiği şekildedir.

$$b_j(x(t)) = P(x(t) = O_j | q_t = q_j), \forall(x(t)), j \quad (1.9)$$

Bu  $x(t)$  fonksiyonu genellikle sürekli zamanlı veriyi iyi tanımlayacak bir dağılımdan seçilmektedir. Tipik bir örnek olarak Gauss Dağılımı gösterilebilir [22].

Bir HMM'e ait başlangıç olasılıkları eşitlik 1.10'da gösterilmiştir.

$$\pi_i = P(q_1 = q_i) \quad (1.10)$$

### 1.2.3. Olabilirlik hesabı

HMM modellerde, Rabiner tarafından tanımlanmış 3 temel problem bulunmaktadır [23]. Bu problemlere ait tanımlar Tablo 1.3'te gösterilmiştir.

Tablo 1.3. Saklı Markov Modellerdeki 3 Temel Problem

Olabilirlik Problemi	Bir HMM ve bir gözlem serisi birlikte verildiğinde ( $\lambda = (A, B, \pi)$ ) ve $O = O_1, O_2, O_3, \dots, O_T$ , bu gözlem serisinin ilgili HMM'den toplam gelme olasılığı nedir? ( $P(O \lambda)=?$ )
Çözümleme Problemi	Bir gözlem serisinin ( $O$ ), bir HMM'den farklı yollar üzerinden gelme durumu olabilir. Bu yollardan maksimum olasılığa sahip yol hangisidir?
Öğrenme Problemi	Bir $O$ gözlem dizisini kullanarak, bir HMM için $A$ ve $B$ parametrelerinin optimize edilmesi.

Burada gerçekleştirilen çalışma kapsamında sadece olabilirlik problemi ile ilgilenildiği için bu bölüm içerisinde olabilirlik problemine odaklanılmıştır.

**Problem Tanımı:** Bir gözlem serisinin bir HMM'e bağlı toplam olabilirliği nedir?

Bir gözlem serisi verildiğinde, bu gözlem serisinin bir durumlar serisine bağlı olasılığı eşitlik 1.11'de gösterilmiştir.

$$P(O|Q) = \prod_{i=1}^T P(O_i|q_i) \quad (1.11)$$

Eşitlik 1.11'e dikkat edilirse, doğrudan durumların görülebildiği varsayımı yapılmıştır. Yani bir t anında gelen bir gözlem değerinin o andaki değeri için hangi durumdan geldiğini doğrudan gözlemlenememektedir. Bu sebepten ötürü her bir t anındaki gözlem değeri için, her bir  $q_i$  değerinin belirli bir olasılık dâhilinde gelebileceğini hesaba katarak işlem yapılması gerekmektedir.

Bu durum Eşitlik 1.12'deki gibi ifade edilebilir.

$$P(O,Q) = P(O|Q) \times P(Q) = \prod_{i=1}^T P(O_i|q_i) \times \prod_{i=1}^T P(q_i|q_{i-1}) \quad (1.12)$$

biçiminde gösterilir. Bu ifade O gözlem dizisinin sadece bir tane Q durumlar dizisinden gelme olasılığını göstermektedir. Eğer O gözlem dizisinin bir HMM içerisindeki bütün durum dizilimlerine göre olasılıklarının toplamının (bir HMM içerisinde bir gözlem serisinin toplam olabirliği) bulunması isteniyorsa Eşitlik 1.13 tüm olası durum dizilimleri için hesaplanarak toplanmalıdır.

$$P(O) = \sum_{\text{tüm } Q} P(O,Q) = \sum_{\text{tüm } Q} P(O|Q) \times P(Q) \quad (1.13)$$

Bu durumda bir O gözlem dizisinin bir HMM'e ait toplam olabirliği elde edilebilir. Fakat bu algoritmanın toplam karmaşıklığı  $O(N^T)$  olup, büyük durum sayıları ve uzun gözlem dizileri için oldukça işlem sayısı fazladır. Dolayısıyla pratikte büyük sayıda durum içeren ve büyük gözlem serileri için olasılık hesabı yapılırken pratikte kullanımı çok uygun değildir.

Bu yöneme alternatif olarak  $O(N^2T)$  karmaşıklığına sahip ileriye yönelik olasılık algoritması (forward algorithm) oldukça kullanışlıdır.

Bu algoritma temelde, bir Saklı Markov Modelini iki parçaya böler. Bu iki parça üzerinden daha verimli bir olasılık hesabı gerçekleştirerek amacına ulaşmayı hedefler.

Bir HMM için, 1'den t anına kadar  $O = O_1, O_2, O_3, \dots, O_t$  gözlemlenmiş olsun. t anında j. durumda bulunulduğu varsayılmış olsun.

Bu duruma ait olasılık değeri:

$$\alpha_t(j) = P(O_1 \dots O_t \mid q_t = q_j) \quad (1.14)$$

ve

$$\alpha_t(j) = \sum_{i=1}^t \alpha_{t-1}(i) a_{ij} b_j(O_t) \quad (1.15)$$

biçimindedir.

Eşitlik 1.14,  $O = O_1, O_2, O_3, \dots, O_t$  geldiği durumda  $q_j$  durumunda olma olasılığının,  $t-1$  anına kadar  $O = O_1, O_2, O_3, \dots, O_{t-1}$  gözlem serisinin geldiği ve  $t-1$  anında  $q_j$  durumunda olma olasılığından, tüm  $j$ . durumlara  $O_t$  gözlem değeri ile geçme olasılıklarının toplamıdır.

$T$  anı son gözlem dizisinin son elmanı ise, bu son eleman için hesaplanan tüm ileriye yönelik olasılık değerlerinin toplamı Eşitlik 1.9'daki gözlem serisinin toplam olasılık değerine eşittir.  $N$  toplam farklı durum sayısı olmak üzere toplam olabirlik değeri 1.16'da gösterilmiştir.

$$P(O|\lambda) = \sum_{i=1}^N \alpha_T(i) \quad (1.16)$$

İleri yönelik olasılık algoritması 3 temel aşamadan oluşmaktadır.

Başlangıç adımı: Bu aşamada  $T$  boyutlu bir gözlem serisinin ilk anındaki gözlem değeri için ileriye yönelik olasılıklar hesaplanır.

$$\alpha_1(j) = \pi_j b_j(O_1) \quad (1.17)$$

Anlaşılabileceği üzere ilk anda bir geçiş olasılığı (transition probability) bulunmamaktadır. Çünkü bir durumdan başka bir duruma geçiş ilk anda söz konusu değildir. Dolayısıyla burada doğrudan başlangıç olasılıkları (initial probability) kullanılmaktadır.

Öz yineleme adımı: 1 anından sonraki tüm ileri yönelik olasılıklar öz yinelemeli (recursive) olarak bir önceki ileriye yönelik olasılık değerlerine bakılarak hesaplanmaktadır.



$$\alpha_t(j) = \sum_{i=1}^t \alpha_{t-1}(i) a_{ij} b_j(O_t), \quad 1 \leq j \leq N, 1 < t \leq T \quad (1.18)$$

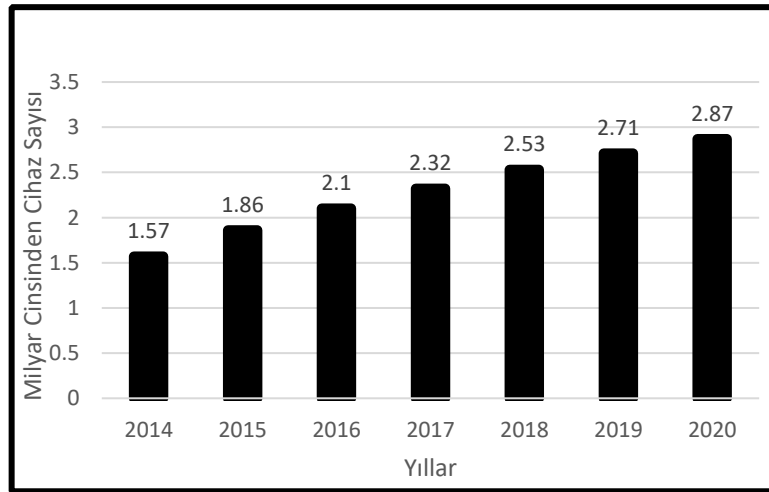
Bu özyineleme adımı gözlem serisinin son elemanı T olmak üzere, T değerine kadar devam eder.

Sonlanma adımı: Sonlanma adımında ise T anındaki ileriye yönelik olasılık değerlerinin toplamı alınır.

$$P(O|\lambda) = \sum_{i=1}^N \alpha_T(i) \quad (1.19)$$

### 1.3. İnsan Aktivite Tespiti ile İlgili Çalışmalar

Statistica'nın yaptığı çalışmaya göre dünya genelinde ortalama mobil telefon sayısının 2018 itibari ile yaklaşık olarak 2,53 milyardır. Bu sayının giderek artarak 2020 yılına kadar toplam 2,87 mobil telefon sayısına ulaşılması beklenmektedir. Mobil cihaz sayısındaki yaygınlaşmanın artması, mobil ortamlarda gerçekleştirilen uygulamaların çoğalmasına ve çeşitlenmesine neden olmuştur.



Şekil 1.4. Dünya genelinde mobil cihaz kullanım sayısı [1]

Bu çalışmalardan biri de insan aktivitelerinin tanımlanmasıdır. İnsan aktivitelerinin tanımlanması problemi, mobil cihazlar için, kişinin üzerinde bulunan mobil bir cihazdaki sensörlerden alınan verilerin anlık olarak işlenerek kişinin o anda gerçekleştirdiği aktivitenin türünün tespiti olarak tanımlanabilir. İnsan aktivitelerinin tespiti, kişiye özel günlük kalorilerin hesaplanması, kişinin gerçekleştirdiği hareketlere

göre sađlık durumunun analizi gibi sađlık uygulamaları olarak veya yařlı insanların buldukları ortamda gerekleřtirdiđi hareketler ile gzetlenmesi, insan pozisyon takibi ve eřitli gvenlik uygulamaları gibi ok eřitli uygulama alanlarında kullanılmaktadır. Bu blmde, 1.1 ve 1.2'de aıklanan yntemler kullanılarak gerekleřtirilen insan aktivitelerinin tanımlanmasına iliřkin literatrde yapılan alıřmalar hakkında bilgi verilmektedir.

### **1.3.1. Sınıflandırma ynemleri ile insan aktivitelerinin tanımlanması**

Literatrde, insan aktivite tanımlanma problemini bir sınıflandırma problemi olarak grp bu sınıflandırma probleminin zm olarak ele alan alıřmalar mevcuttur. Bu alıřmalar temelinde her bir insan aktivitesini farklı sınıf etiketleri olarak alarak, sensr verilerinden elde ettikleri znelikler ile bu sınıf etiketlerinin tespiti zerine kuruludur.

Bao ve Intille, 5 tek eksenli ivmeleri, kala, el ve ayak bilekleri gibi insan vcudunun farklı yerlerine yerleřtirerek toplam 20 farklı denekten veri toplamıřlardır [24]. 20 farklı denekten topladıkları 20 farklı insan aktivitesini, Fourier Dnřm zneliklerini kullanarak, rnek tabanlı đrenme, karar ađaları, Naive Bayes sınıflandırıcılar ile eđitmiřler ve %80'e yakın bir bařarım elde etmiřlerdir. Benzer řekilde Banos ve arkadařları da vcudun farklı blmlerine yerleřtirilmiř ivmelerler kullanılarak, yrme, kořma, ayakta durma ve oturma hareketlerini tespit etmeye alıřmıřlardır [25].

Niskham ve Nikhil, -eksenli bir ivmeler kullanarak, yrme, kořma, ayakta durma, oturma, merdiven inme, merdiven ıkma, diř firalama ve sprme hareketlerini tespit etmeye alıřmıřlardır [26]. ođunluk oylamasına dayalı sınıflandırıcılarının diđer sınıflandırıcılardan daha bařarılı olduđu sonucuna varmıřlardır. Ayrıca diř firalama ile merdiven inme aktivitelerinin birbirinden ayırt edilebilirliđinin imknsız olduđunu belirtmiřlerdir.

Kwapsiz ve arkadařları bir android cihaz zerinden aldıkları -eksenli ivmeler verilerini kullanarak bir aktivite tespiti yapmıřlardır [27, 28]. Ortalama ivme, standart sapma, ortalama mutlak sapma gibi farklı zellikler ıkarmıřlardır. Elde ettikleri veri setini 3 farklı sınıflandırıcının eđitimi (Lojistik Regresyon, Karar Ađacı ve MLP) iin kullanmıřlardır. Karar Ađacının modellerinin diđer modellerden bařarımının daha yksek olduđunu belirtmiřlerdir. Elde ettikleri sonularda merdiven inme ve merdiven ıkma hareketleri arasında sınıflar arası benzerlik sebebiyle bařarımın dřk olduđu gzlemlenmektedir.

[29]'da, zaman ve frekans uzayında olmak üzere toplamda 13 farklı dalga özniteliği kullanarak, 5 farklı insan aktivitesi tespit edilmeye çalışılmıştır. Veri seti 27-35 yaş arasından 2 kişi kadın olmak üzere toplam 7 farklı denekten toplanmıştır. 69400 adet sensör verisinden 13 farklı zaman ve frekans özniteliği elde ettikten sonra C4.5 algoritması ve Yapay Sinir Ağları ile sistemlerini eğitmişlerdir. Elde ettikleri sonuçlara göre C4.5 algoritması ile %94,13'lük bir başarı oranı elde etmişlerdir.

Wu ve ark., ivmeölçer verisiyle birlikte jiroskop verilerini kendi insan aktivite sistemleri için kullanmışlardır [30]. Elde ettikleri sonuçlara göre ivmeölçer verileri ile birlikte kullanıldığında jiroskop verileri daha başarılı modellerin kurulmasında katkıda bulunmaktadır. Benzer şekilde [31]'de yapılan çalışmada da ivmeölçer ve jiroskop sensörlerini ve ek olarak GPS verilerini birlikte kullanarak 7 farklı aktivite için bir tespit sistemi sunulmuştur. Literatürdeki çoğu çalışma gibi burada da Karar Ağacı algoritması insan aktivite tespit sisteminde en iyi sonucu elde etmiştir.

### **1.3.2. Saklı Markov modelleri ile insan aktivitelerinin tanımlanması**

Sınıflandırma problemi dışında, bir insan aktivitesini örüntü tanıma problemine dönüştürerek tanımlamaya çalışan çalışmalarda literatürde mevcuttur. Bu türden çalışmalar temelde, farklı insan aktivitelerini farklı örüntüler olarak kabul ederek bu örüntülerin tanınması esasına dayanmaktadır.

Lee ve Cho [32], kurdukları iki aşamalı HMM yapısında öncelikle aksiyon olarak belirttikleri oturma, yürüme, merdiven hareketleri ve koşma hareketlerini tespit etmişlerdir. Elde ettikleri aksiyonlar üzerine kurulu ikinci bir HMM model ile asıl amaçları olan alış-veriş yapmak, otobüs ile seyahat etmek ve hareketli durum olarak adlandırdıkları farklı insan hareketlerini tespit etmeye çalışmışlardır.

Kim ve arkadaşları, Saklı Markov Modellerinin Birleşimi (Hidden Markov Model Ensemble – HMME) metodunu kullanmışlardır [33]. 3-eksenli ivmeölçer verileri üzerinde özellik çıkarımı uygulamışlardır. Bu veriden farklı aktiviteler için oluşturdukları HMM modelleri kullanarak Karar Profilleri (Decision Profile – DP) elde etmişlerdir. Elde ettikleri DP'ları her bir aktivite için oluşturulmuş Karar Şablonlarıyla (Decision Templates – DTs) ile benzerliklerini ölçmüş ve bir hareketin türünü tespit etmişlerdir. Sundukları yöntemi farklı sınıflandırıcılar ile karşılaştırmışlar ve MLP'dan sonra en başarılı model olduğunu tespit etmişlerdir.

Ranao ve Cho yaptıkları çalışmalarda 2 aşamalı bir aktivite tespit sistemi sunmuşlardır [3, 34]. İvme verisi dışında geliştirdikleri sistemde jiroskop verilerini de kullanmışlardır. Kaba sınıflandırma olarak adlandırdıkları birinci aşamada bir hareketin durağan (stationary) veya hareketli (moving) bir eylem olduğunu tespit etmeye çalışmışlardır. Hareket çeşidinin durağan veya hareketli olmasına göre ikinci aşamada iki farklı HMM'den birini kullanarak aktivitenin gerçek sınıfını tespit etmişlerdir.

Han ve ark., geri yönlü geçişlere veya geçişlerin es geçilmesine izin veren çok yönlü bir HMM ile insan aktivite tanımlama işlemlerini gerçekleştirmişlerdir [35]. Ayrıca yerçekiminden kaynaklı sapmayı (bias) hesaba katmışlar ve verideki bu durumu telafi etmeye çalışmışlardır. Elde edilmiş eğimi düzeltilmiş (tilt compensated) veri üzerinden sistemlerini gerçekleştirmişlerdir. Sonuç olarak eğim düzeltme tekniklerinin sensörün rotasyonu karşısında güçlü bir model olduğunu ve sundukları çok yönlü HMM modelin soldan sağa tek yönlü bir HMM'den daha başarılı olduğu sonucuna varmışlardır.

[36]'da yaptıkları çalışmada Mannini ve Sabatini, Bao ve Intille'ye ait [24]'deki çalışmalarında kullandıkları veri setini kullanarak, 7 farklı temel insan aktivitesi için, Sürekli Zamanlı Saklı Markov Modelleri ile oldukça yüksek bir başarıma ulaşmışlardır.

## **2. İNSAN AKTİVİTE TESPİTİ İÇİN SİSTEM MİMARİSİ**

İlgili tez çalışması kapsamında geliştirilen iki farklı insan aktivitesi tanımlama (Human Activity Recognition – HAR) sistemi geliştirilmiş ve karşılaştırılmıştır. İlerleyen bölümlerde bahsedileceği üzere bu iki sistemin temel aşamaları aynıdır. Fakat iki sistem arasındaki temel farklılık öznitelik çıkarımı esnasında karşımıza çıkmaktadır. İlgili çalışmaların temel amacı bir mobil cihaz üzerinden toplanan bir kişiye ait verileri kullanarak gerçek zamanlı bir insan aktivite tespit sistemi gerçekleştirilmesidir. Bu bölüm altında ilgili sistemlerin altyapısal tasarımları ve iş akışları hakkında detaylı bilgiler verilmiştir. İlerleyen bölümlerde bu sistemler ile alakalı olarak gerçekleştirilen deneyler ve bu deneylerin karşılaştırmaları yer almaktadır.

### **2.1. İnsan Aktivitesi Tanımlama Aşamaları**

Mobil cihazlar üzerinden gerçekleştirilen ve veri madenciliği yöntemlerinin kullanıldığı bir HAR sistemine ait genel tasarımsal şema esasında standart bir veri madenciliği problemi olan sınıflandırma problemi olarak ele alınabilir. Bir sınıflandırma problemi olarak ele alındığında bir HAR sistemi, esasında bir hareket türünü temsil eden veri noktalarının temsil ettiği hareket türünün tespiti olarak adlandırılabilir. Temel amaç bir kişi üzerine yerleştirilmiş olan mobil bir cihazdaki bulunan farklı sensör (ivme-ölçer, jiroskop vb.) türlerinden elde edilen verilerin yorumlanarak ilgili hareketin gerçekte ne olduğunun bulunması ile alakalıdır.

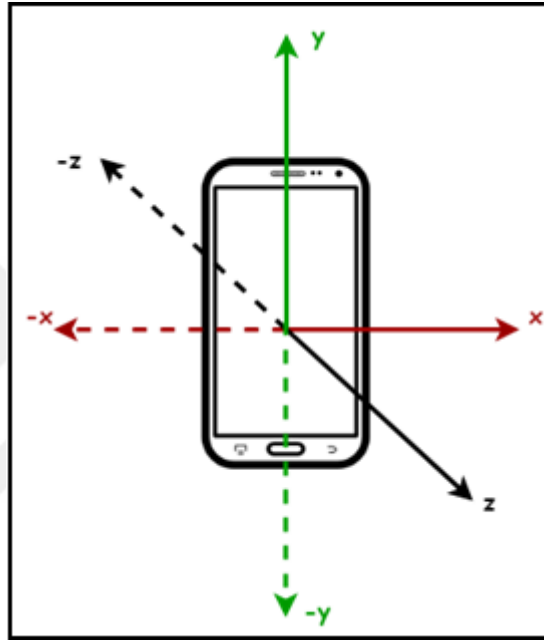
Bunula alakalı olarak bir HAR sistemi temel veri madenciliği adımlarını içermektedir. Bu adımlar veri toplanması, veri temizliği, verilerin kullanılarak bir HAR sistemi için daha fazla anlam ifade edecek yeni özniteliklerin bulunması ve bu özniteliklerin kullanılarak farklı sınıflandırma algoritmalarıyla bir HAR modelinin kurulması olarak sıralanabilir. Elde edilen farklı sınıflandırma modellerinin sonuçları yorumlanarak en başarılı modelin bulunması temel amaçtır.

#### **2.1.1. Veri toplama**

Bu başlık altında bir mobil cihaz üzerindeki ivme-ölçer verisi kullanılarak nasıl veri toplandığı anlatılmaktadır.

Bir HAR sistemi gerçekleştirilirken ilk adım veri toplama aşamasıdır. Burada verilen örnek bir mobil telefon üzerindeki 3-ekseli bir ivmeölçer kullanarak insan aktivitelerinin toplanması ile alakalıdır.

3-eksensli bir ivmeölçer bulunduğu 3-boyutlu uzaydan anlık olarak ivmelenme bilgilerini toplayan bir sensör türüdür. Şekil 2.1'de bir mobil telefonda bulunan 3-eksensli ivmeölçere ait eksenler gösterilmiştir. 3-eksensli ivme ölçer barındıran bir mobil telefon örneği gösterilmiştir.



Şekil 2.1. İvmeölçere ait eksenler

Yalnızca ivmeölçer verisi kabul edilirse, fiziksel anlamda bir insan aktivitesi 3 farklı ekseninde gerçekleşen ivmeli bir harekettir. Örneğin yürüme hareketinde bir insan biyolojik yapısı gereğince ileriye doğru bir ivmelenme hareketi (z-ekseni yönünde) yapmaktadır. Yürüme esnasında temel olarak ileriye doğru hareket edilse de aslında bacaklara ait eklem noktalarının farklı açılarda hareket etmesinden dolayı yukarı-aşağı yönlerde (y-ekseni yönünde) de bir hareket gerçekleştirmektedir. Bu hareketlerdeki, sola ve sağa doğru ivmelenme ise x-ekseni ile temsil edilmektedir. Yerden zıplama hareketi ise temel anlamda yukarı-aşağı ekseninde bir hareketin gerçekleştiği bir hareket türü olarak açıklanabilir. Zıplama hareketi ile yürüme hareketi arasındaki en karakteristik fark z-ekseni yani ileriye doğru alınan ivme cinsinden farklılık göstermektedir. Bir başka hareket türü olan koşma hareketi ise yürüme ile yakın olmasına rağmen daha hızlı bir hareket olduğundan dolayı aynı eksenlerdeki ivme değişimleri büyük farklılıklar içermektedir. İşte bu türden farklılıkları kullanarak

(ivmelenme yönleri, ivmelenme şiddetleri vb.) bir HAR sistemi insan hareketlerini birbirinden ayırmaya çalışmaktadır.

Bu farklılıkların tespit edilmesi için öncelikli olarak farklı hareket türlerine ait yeterli miktarda verinin toplanması gerekmektedir. Veri toplama aşamasında farklı kişiler üzerine yerleştirilmiş 3-eksenli ivmeölçere sahip mobil cihazlar kullanılarak bu kişilerden farklı türden hareketleri gerçekleştirilmesi istenmektedir. Kişiler ilgili hareket türlerini gerçekleştirdiğinde bu hareketin türüne ait bilgi (etiket verisi) ve harekete ait ivme değerleri gerçek zamanlı toplanarak farklı hareketlere ait veri noktaları dış dünyadan toplanmaktadır.

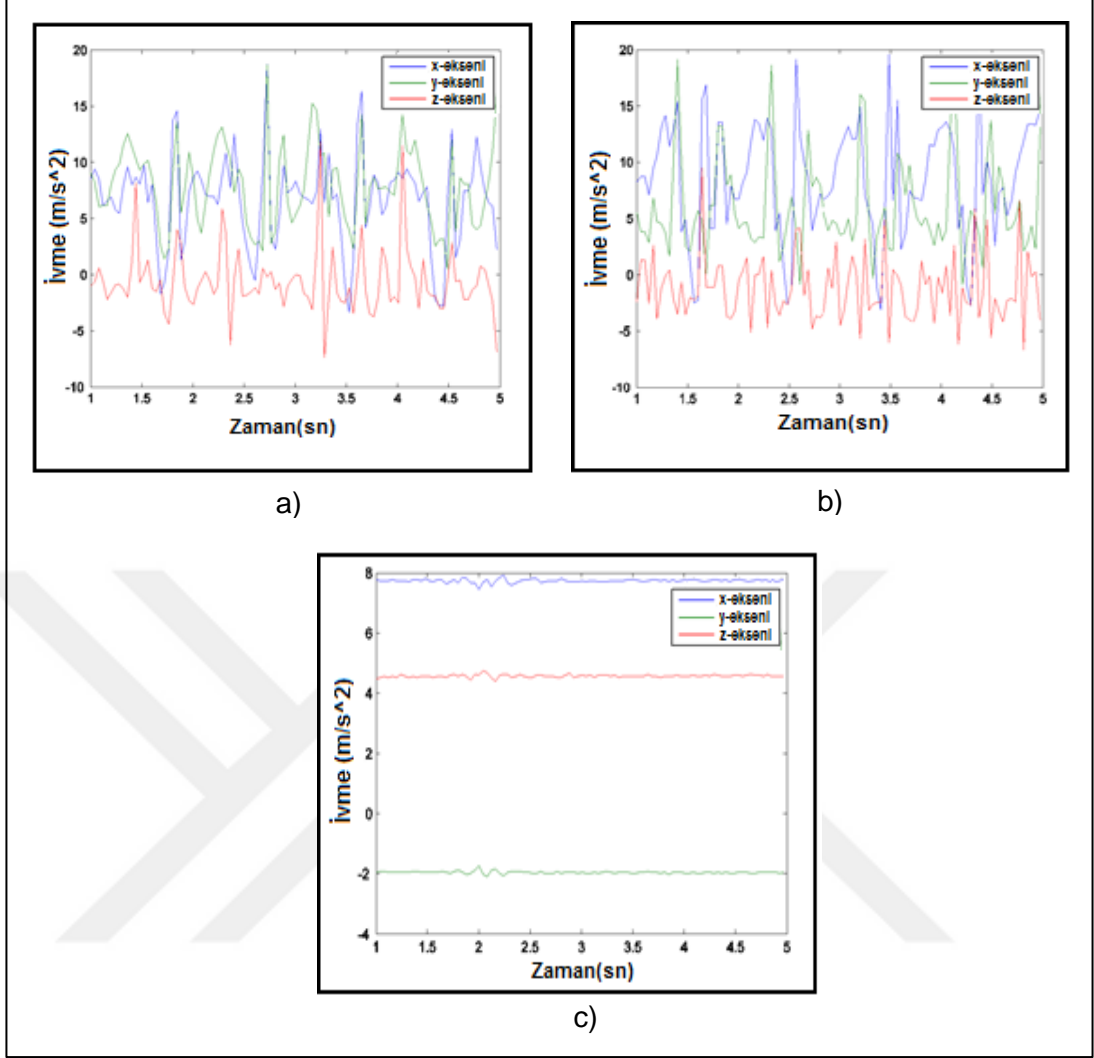
Toplanan veri esasında, sürekli zamanlı bir dalga yapısı taşımaktadır. Bir harekete ait gözlem dizisini Eşitlik 2.1'deki gibi gösterebiliriz.

$$O_{1:T} = \{O_{1:T}^x, O_{1:T}^y, O_{1:T}^z\} \quad (2.1)$$

Burada 1:T ilgili gözlem serisine ait toplam kaydedilen süreyi, x-y-z ise 3-farklı eksenini temsil etmektedir.

5 saniyelik bir zaman aralığı için, 50 Hertz (Hz) frekansında toplanmış farklı hareket türlerine ait sürekli zamanlı ivme değerleri Şekil 2.2'de gösterilmiştir. Şekil 2.2'de görülebileceği üzere farklı hareket türlerine ait ivme değerleri farklılık göstermektedir. Örneğin Merdiven çıkma hareketi için z eksenindeki ivme değerlerinin periyodu yürüme hareketine göre daha fazladır. Y eksenini için yürüme hareketindeki değişim miktarı merdiven çıkma hareketine ait y eksenindeki değişim miktarından daha fazladır. Veya oturma hareketi için düşünecek olursak tüm eksenlerdeki ivme değerlerin beklenebileceği gibi durağan bir yapı sergilemektedir.

Bir insan hareketi yukarıda açıklandığı gibi sürekli zamanlı bir aktivite olduğundan dolayı standart veri madenciliği yöntemleri bu türden bir veri için doğrudan uygulanamaz [28]. Bu sebepten dolayı verilerin öncelikle ayrık zamanlı verilere dönüştürülmesi ve ayrıklaştırılmış bu veri üzerinde veri madenciliği algoritmalarının çalıştırılması gerekmektedir [37]. Ayrıca sadece 3-eksendeki ivme değerlerinden doğrudan türetilmiş özniteliklerin kullanılması yeterli olamayabilir. Bu sebeple ivme verileri üzerinde öznitelik çıkarımı (feature extraction) yapılması gerekmektedir.



Şekil 2.2. Gerçek zamanlı hareketlere ait ivme değerleri: a) Yürüme b) Merdiven İnme c) Merdiven Çıkma

## 2.2. Öznitelik Çıkarımı

Veri seti elde edildikten sonra bir hareketi en iyi ifade edecek özniteliklerin bulunması gerekmektedir. Dolayısıyla bu özniteliklerin keşfedilmesi modelin başarımı ile doğrudan alakalıdır. Buna bağlı olarak 3-eksenli ivmeölçer verilerini kullanarak gerçekleştirilen iki farklı öznitelik çıkarımı uygulanmıştır. Bu bölümde, Bölüm 3'teki deneysel çalışmalar altında kullanılmış olan öznitelik çıkarımı aşamaları açıklanmıştır.

### 2.2.1. Sürekli zamanlı dalga özniteliklerinin kullanılması

2.1'de sürekli zamanlı verilerin standart veri madenciliğine uygulanamadığı ve veriler üzerinde ayırıklaştırma yapılmasının gerekliliği tartışılmıştır.



Dolayısıyla ivmeölçer üzerinden elde edilmiş sürekli zamanlı verilerinin ayrıklaştırılması gerekmektedir. İvmeölçer benzeri sensörler tasarımları gereği esasında farklı zaman aralıklarıyla (periyotlarla) veri toplayabilmektedir. Bu sebeple esasında bu veriler zaten ayrık durumdadır. Fakat bu türden veriler için sürekli zamanlı oldukları varsayımı yapılmıştır. İlk öznitelik çıkarımı yönteminde sürekli zamanlı dalga özelliklerinden faydalanılmıştır [38].

Bu tez kapsamında kurulmuş olan HAR sistemi, hareketi gerçekleştiren kişiye ait sınırlı bir zaman aralığında gerçekleştirilmiş hareketi tahmin etmeye çalışmaktadır. Örneğin 10 saniyelik bir hareket eyleminin ne olduğunun tahmin edilmesi problemin tanımı olabilir. Dolayısıyla sürekli zamanlı dalga özellikleri elde edilirken verinin tamamının kullanılması yerine tahmin edilecek periyota eşit aralıklara bölünmesi ve bu aralıklara ait özniteliklerin çıkarılması gerekmektedir. Dolayısıyla bu öznitelikler belirli bir periyot içerisindeki sürekli zaman öznitelikleri olarak düşünülebilir.

Bu kapsamda 1:T arasındaki bir sürekli zamanlı dalga kesitine ait öznitelikle, k ilgili eksen olmak üzere aşağıda gösterilmiştir:

Her eksen için ortalama ivme (3-öznitelik):

$$a_{ort}^k = \frac{1}{T} \sum_{i=1}^T a_i^k \quad (2.2)$$

Her eksen için standart sapma (3-öznitelik):

$$\sigma^k = \sqrt{\sum_{i=1}^T \frac{(a_i^k - a_{ort}^k)^2}{T}} \quad (2.3)$$

Ortalama toplam ivme (1-öznitelik):

$$a_{ort} = \sqrt{(a_{ort}^x)^2 + (a_{ort}^y)^2 + (a_{ort}^z)^2} \quad (2.4)$$

Bu öznitelikler 1:T arasındaki bir insan aktivitesi için ivmelerin olasılıksal dağılımını ifade etmektedir. Belirli sınır ivme değerleri (binler) ile ivmeler için farklı gruplar oluşturulur. Bu sınır değerleri arasında kalan ivmelerin tüm ivme değerlerine oranı bir öznitelik olarak tutulmaktadır. Her eksen için toplam 10 adet olmak üzere, 3-eksende

toplam 30 adet hesaplanan gruplanmış dağılım değerleri öznitelik olarak saklanmaktadır. Eşitlik 2.5'te bu durumun matematiksel ifadesi gösterilmiştir.

Gruplanmış Dağılım (Binned Distrubiton) – (30 öznitelik):

$$P(\text{bin}_j^k) = \frac{1}{T} \sum_{i=1}^T R(a_i^k), \quad R(a_i^k) = \begin{cases} 1, & \text{eğer } a_i^k \text{ bin}_j^k \text{ sınırları içinde ise} \\ 0, & \text{diğer durumlarda} \end{cases} \quad (2.5)$$

Eşitlik 2.5'te k. eksenindeki j. grup için olasılık dağılımı hesaplanmaktadır.

Her eksen için Ortalama Mutlak Sapma (MAD) – (3 öznitelik):

$$\text{MAD}^k = \frac{\sum_{i=1}^T |a_i^k - a_{\text{ort}}^k|}{T} \quad (2.6)$$

Sürekli zaman özellikler hesaplandığında ham veri ayrıklaştırılmıştır. Öznitelik çıkarım işlemi sonucunda toplam 40 öznitelik hesaplanmaktadır. Bu özniteliklere ek olarak hareket türünü belirten bir etiket bilgisi de yer almaktadır. Ayrıklaştırılan etiketli veri sınıflandırma algoritmalarına giriş olabilecek nitelik taşıyor hale gelmiştir.

## 2.2.2. Olasılıksal özniteliklerin kullanılması

1.3'te açıklanan Saklı Markov Modellerinin (HMM), zincirleme olarak gözlemlenen bir dizi olayın, doğrudan gözlemleyemediğimiz saklı durumların bir araya gelerek oluşturduğu olgusuna dayandığından bahsedilmiştir. Belirtildiği üzere HMM yapısı doğası gereği örüntü tanıma problemlerinde sıklıkla kullanılmaktadır. Bir örüntü, aslında bir dizi kural veya kurallara bağlı, birbirleriyle ilişkisi bulunan olaylar dizisinin gerçekleşmesinin doğal bir sonucudur. Bir insan aktivitesi (hareketi) tahmin edilebileceği gibi temelde bir örüntüdür. Örneğin bir insanın gerçekleştirdiği yürüme, koşma, oturma gibi eylemler kendi içerisinde benzerlik taşımaktadır ve biz insanlar bu olguları birbirinden bu sayede ayırt edebiliyoruz. Dolayısıyla bir insan aktivitesi kolaylıkla bir örüntü tanıma problemi haline gelebilir. Matematiksel anlamda bu ifade, farklı örüntüleri tanıyabilen farklı modellerin kurulması ve gelen bir örüntünün farklı örüntü modellerinin hangisine benzediğinin tespit edilmesi olarak ifade edilebilir.

HMM'ler açısından inceleyecek olursak, bu örüntü tanıma problemi her bir örüntüye ait farklı HMM'lerin kurulması ve gelen bir örüntünün bu HMM'lerden olabilirlik (likelihood) açısından hangisine daha çok benzediğinin tespiti şeklinde gerçekleşmektedir.

Bu konu ile bağlantılı olarak bir insan aktivitesi tanımlama probleminde, farklı insan aktivitelerine ait farklı HMM'ler kurulabilir. Dolayısıyla dış dünyadan alınan belli bir periyoda sahip insan hareketinin hangi insan aktivitesine olasılıksal olarak ne kadar benzediği tespit edilebilir.

Bu tez çalışmasında literatürdeki HMM model kullanan çalışmalardan farklı olarak, HMM modellerin doğrudan sınıflandırma amacıyla kullanmak yerine öznitelik olarak kullanılmıştır. Bu şekilde bir yöntem izlenmesinde iki temel amaç bulunmaktadır. İlk olarak 2.2.1'de açıklandığı üzere "Gruplanmış Dağılım" özniteliği, 1:T zaman aralığındaki bir insan aktivitesinin ivmelerinin olasılıksal dağılımını göstermektedir. Bu bakış açısı ivmelerin olasılıksal dağılımı yerine, bu ivme dizisinin farklı insan hareketlerine benzerliklerinin olasılıksal olarak dağılımı nasıl olmaktadır sorusuna dönüştürülmesi olarak ortaya çıkmıştır. İkinci olarak, ilgili ivme dizisine ait farklı insan aktivitelerinden olabilirliklerinin korelasyonu bir anlam ifade edilebilir mi sorusu bu çalışmanın temel motivasyonunu oluşturmaktadır.

Bir insan aktivitesi daha önce çok kez belirtildiği şekilde, çok boyutlu (bizim çalışmamızda 3 boyutlu uzay) bir ortamda gerçekleşen bir sürekli zamanlı dalgadır. Dolayısıyla HMM'ler oluşturulurken bu 3-boyutlu uzayın birleşik bir şekilde veya ayrı ayrı her boyut için bir örüntü yapısı olmaktadır. Bu çalışmada her bir eksen farklı bir örüntü olarak alınmış ve her eksene ait HMM'ler oluşturulmuştur.

Bir insan aktivitesi için act aktivite türü, x-y-z ilgili eksen olmak üzere, bir HMM:

$$P_{act}(O|\lambda) = \{P_{act}(O_{1:T}^x|\lambda^x), P_{act}(O_{1:T}^y|\lambda^y), P_{act}(O_{1:T}^z|\lambda^z)\} \quad (2.7)$$

şeklinde gösterilmektedir.

Gerçek zamanlı bir hareket için, gelen harekete ait sürekli zamanlı ivme dizisi alınarak her bir insan aktivitesine ve bu aktiviteye ait oluşturulmuş 3 eksene ait HMM'lere giriş olarak verilmektedir. Elde edilen çıktılar hareketi tanımlayacak öznitelik uzayındaki öznitelikler şeklinde tutulmaktadır. HMM öznitelikleri dışında daha önce 2.2.1'de açıklanmış olan ve her eksende hesaplanmış, ortalama ivme, standart sapma, ortalama mutlak sapma ve ortalama ivmeler kullanılarak oluşturulmuş ortalama toplam ivme öznitelikleri hesaplanmıştır. Bu özniteliklere ek olarak 2.2.1'deki gruplanmış dağılım yerine olasılıksal öznitelikler olarak adlandırabileceğimiz HMM'lere ait ileriye yönelik olasılık (forward probability) değerleri, her hareket

çeşidine ait 3 eksen de olacak şekilde hesaplanarak öznitelik uzayını arttıracak şekilde eklenmiştir.

Bu şekilde bir öznitelik uzayı oluşturarak gruplanmış dağılım ile olasılıksal özniteliklerin bir insan aktivite tanıma sisteminde etkisini karşılaştırabilecek bir altyapıyı da oluşturmuş oluyoruz.

Sonuç olarak Bölüm 3 Deneysel Çalışmalar altında detaylı açıklaması yapılacak olan 6 farklı harekete ilişkin (yürüme, koşma, merdiven inme, merdiven çıkma, ayakta durma, oturma) 3 eksen de toplam 18 farklı HMM'in de bulunduğu öznitelik uzayı Tablo 2.1'de gösterilmiştir.

Tablo 2.1. Olasılıksal öznitelikler kullanan veri seti

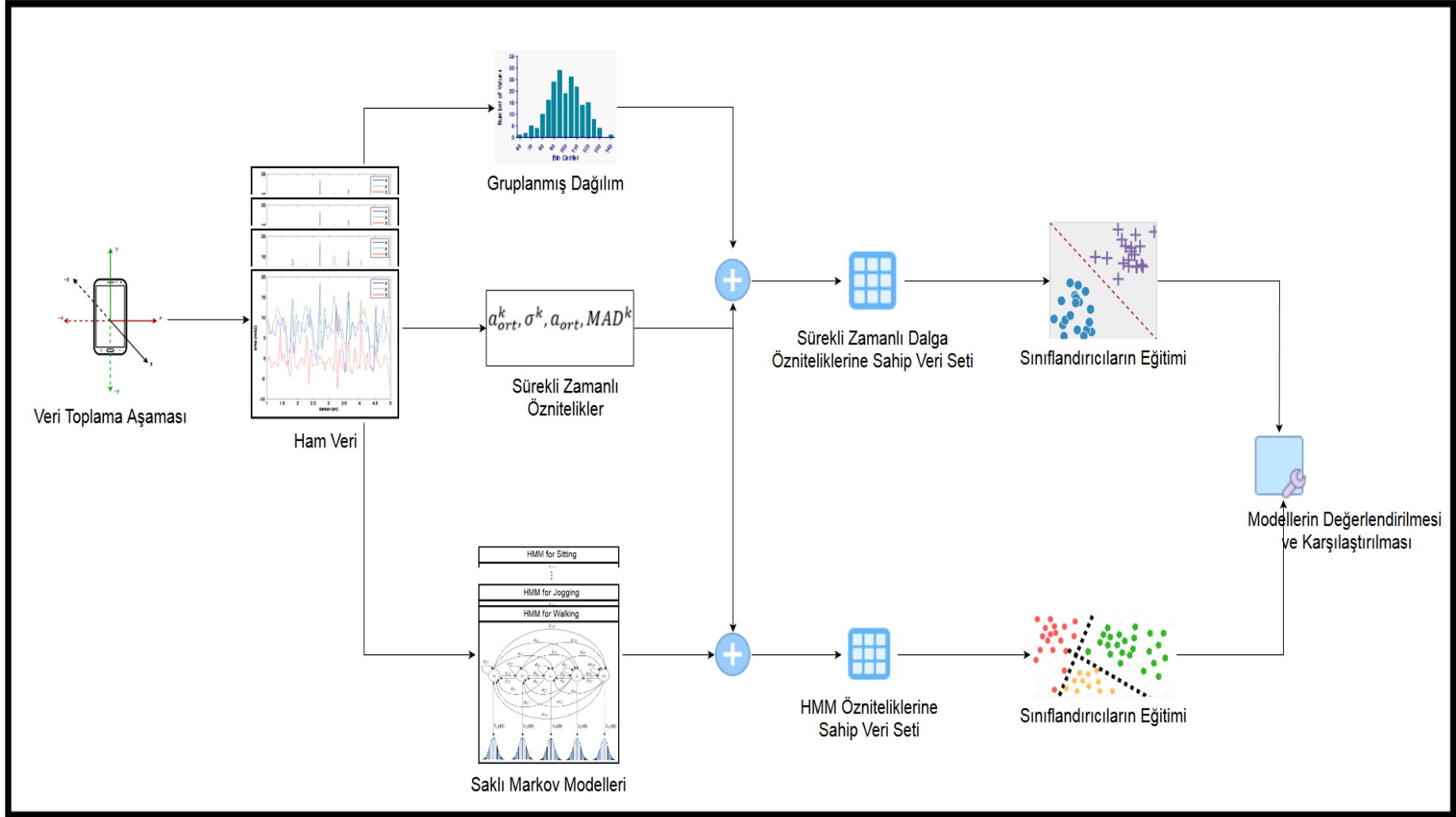
Ortalama ivme değeri (3 öznitelik)	Her eksen için ortalama ivme
Standart Sapma (3 öznitelik)	Her eksen için standart sapma
Ortalama toplam ivme (1 öznitelik)	$\sqrt{a_x^2+a_y^2+a_z^2}$
Ortalama mutlak sapma (3 öznitelik)	Her eksen için, gözlem serisinde bulunan ivme değerlerinin ilgili eksen de ortalama ivmeden toplam sapma miktarı
HMM Öznitelikleri (18 öznitelik)	Gözlem serisine ait 3-eksen de farklı hareket türlerinden gelme olasılıkları
Sınıf etiketi	Hareket türü

Yukarıda açıklanan şekilde veri seti ayrıklaştırıldıktan sonra artık etiketli veri sınıflandırma algoritmalarına giriş oluşturacak yapıya gelmektedir.

### 2.3. Sistem Dizaynı

Şekil 2.3'te bu tez çalışması çerçevesinde gerçekleştirilmiş olan insan aktivite tespit sistemi için gerçekleştirilen temel aşamalar gösterilmiştir. Bu bölümde gerçekleştirilen aşamalar Şekil 2.3'te belirtilen farklı iki veri setinin oluşturulması ile alakalı olan aşamalara kadardır. Bir sınıflandırma problemi olarak ele alınmış olan farklı insan aktivitelerine ait yeteri kadar verinin toplanması gerekmektedir. Bu çalışmada, Bölüm 3 Deneysel Çalışmalar kısmında açıklanacağı üzere hazır bir insan aktivite veri seti kullanılmıştır. Veri setine ilişkin ayrıntılar bu bölümde verilmiştir.

Her eksen için: ortalama ivme, standart sapma ve ortalama mutlak sapma değerleri gruplanmış dağılım özniteliklerinin kullanıldığı bir veri seti ile HMM öznitelikleri kullanılmak üzere toplam iki farklı veri seti oluşturulmuştur. Veri seti oluşturulduktan sonra gerçekleştirilen aşamalar artık bir sınıflandırma problemine dönüşmektedir. Bu aşamalar Bölüm 3'te açıklanmıştır.



Şekil 2.3. İnsan Aktivite Tespit Sistemi için Gerçekleştirilen Aşamalar

### 3. DENEYSEL ÇALIŞMALAR

Çalışma kapsamında, Bölüm 2.2.1 ve Bölüm 2.2.2'de bahsedilen öznelikler kullanılarak iki farklı veri seti üzerinden aktivite tespit sistemi gerçekleştirilmiştir. Farklı sınıflandırma algoritmaları ile farklı modeller eğitilmiştir. Öncelikle veri setleri için oluşturulan modeller kendi içlerinde analize edilmiştir. Ek olarak farklı özneliklerin etkisinin karşılaştırılması amacıyla ham veri setinden oluşturulan iki farklı veri setine ait sonuçlar da karşılaştırılmıştır.

#### 3.1. Veri Setlerinin Hazırlanması

##### 3.1.1. Ham veri seti

Çalışma kapsamında Wireless Sensor Data Mining (WISDM) ekibinin insan aktivite veri seti kullanılmıştır [18]. Veri seti toplam 36 farklı deneğin ön cebine yerleştirilmiş 20Hz frekansında 50ms'lik zaman aralığında ölçüm yapan bir akıllı telefon tarafından toplanmıştır. Kullanıcılardan yürüme, koşma, oturma, ayakta durma, merdiven inme ve merdiven çıkmak üzere toplam altı farklı günlük insan aktivitesi gerçekleştirilmesi istenerek ilgili veriler toplanmıştır.

Veri seti 6 farklı insan aktivitesinden 50ms periyotla toplam 1.098.207 adet farklı kayıt almıştır. Bu kayıtların dağılımı Tablo 3.1'de gösterilmiştir.

Tablo 3.1. Veri setindeki insan aktivitelerinin dağılımı

İnsan Aktivitesi	Kayıt Sayısı	Dağılım(%)
Yürüme	424.400	%38,6
Koşma	342.177	%31,2
Merdiven Çıkma	122.869	%11,2
Merdiven İnme	100.427	%9,1
Oturma	59.939	%5,5
Ayakta Durma	48.395	%4,4

Ham veri içerisinde bir insan aktivitesine ait toplam 6 farklı öznelik bulunmaktadır. Bu öznelikler Tablo 3.2'de gösterilmiştir.

Tablo 3.2 Veri setindeki kayıtlara ait öznelikler

denek_no	aktivite_tipi	zaman damgası	x-ivmesi	y-ivmesi	z-ivmesi
----------	---------------	---------------	----------	----------	----------

Bu özneliklere karşılık gelen açıklamalar aşağıda açıklanmıştır.

- denek\_no: Hareketi yapan kişinin kaç numaralı denek olduğunu gösterir.
- aktivite\_tipi: İlgili deneğin gerçekleştirdiği hareketin türü (yürüme, koşma, oturma vs.)
- Zaman Damgası: İlgili kişiden veri bir telefon uygulaması aracılığıyla toplanmıştır. Kullanıcıdan alınan verinin uygulama başladıktan ne kadar süre sonra alındığının nano-saniye cinsinden değeridir.
- x-ivmesi: İlgili deneğin gerçekleştirdiği insan aktivitesine ilişkin x-ekseninden alınan anlık ve  $[-20,20]$   $m/s^2$  aralığında toplanmıştır. Burada ivmenin, 0 değerinde bulunması kişi üzerinde hiçbir ivme değerinin etki etmediğine,  $9,81 m/s^2$  ise kişiye sadece yerçekimi ivmesinin etki ettiğine işaret etmektedir. Dolayısıyla eğer telefon düz bir yüzeyde ise veya kişi hareketi durağan olan bir insan aktivitesi gerçekleştiriyorsa (oturma ve ayakta durma hareketleri), bu durumda bazı eksenlerde etkiyen ivme değeri  $1g$  değerine yani  $9,81 m/s^2$  değerine yakın bir değerdir.
- y-ivmesi: y-ekseninde elde edilen ivme değeridir. Konuya ilişkin detaylı açıklama x-ivmesinde yapılmıştır.
- z-ivmesi: z-ekseninde elde edilen ivme değeridir. Konuya ilişkin detaylı açıklama x-ivmesinde yapılmıştır.

### 3.1.2. Sürekli zamanlı dalga öznitelikleri

3.1’de bahsedilen ham veri setinin Bölüm 2.1’de açıklandığı üzere ayrıklaştırılması gerekmektedir. Ve bu çalışmada “Sürekli Zamanlı Dalga Özelliklerini” kullanarak bir öznitelik çıkarımı ile bir ivme serisinin eksenleri için farklı hareket türlerinden gelme olasılıklarını hesapladığımızı ikincil bir “HMM tabanlı bir öznitelik çıkarımı” aşamaları uyguluyoruz.

İlk olarak sürekli zamanlı dalga öznitelikleri çıkartılmıştır. Öncelikli olarak ivme değerleri 10 saniyelik kayıtlar halinde okunarak ivme serisi haline getirilmiştir. Okunan her bir ivme serisi için Bölüm 2.2.1’de açıklanan öznitelikler hesaplanmıştır. Sonuç olarak 40 öznitelik ve 1 sınıf etiketinden oluşan kesikli zamanlı bir veri seti oluşturulmuştur.

### 3.1.3. HMM (olasılıksal) öznitelikleri

Bölüm 2.2.2’de açıklandığı üzere bir insan aktivitesinin olasılıksal olarak farklı hareketlere farklı olasılık değerlerinde benzeyebileceğini ve bu olasılık değerlerinin bir arada incelenmesinin ilgili hareket türüne ait bir anlam ifade edebileceği gerçeği

üzerinde durulmuştur. Dolayısıyla bu bölümde, Bölüm 2.2.2'de bahsedilen farklı hareket türleri için HMM modellerinin nasıl kurulduğuna ilişkin detaylar yer almaktadır.

Burada standart bir yapı istediğimizden dolayı tüm HMM modellerin özdeş prensipler üzerine kurulmasına karar verilmiştir. Hatırlanacağı üzere her eksendeki ivme değerlerinin Gruplanmış Dağılımları çıkartılmıştır. Her eksen de ivme değerleri toplam 10 parçaya bölünmüş ve 3-eksende 30 adet grup (bin) değeri çıkartılmıştır. HMM tabanlı sistem oluştururken bu grup yapıları, HMM durumları (states) olarak belirlenmiştir.

Bu şekilde bir yordam izlenmesinin bazı sebepleri vardır. Daha önce belirtildiği üzere bir insan aktivitesinin doğası aslında sürekli zamanda gerçekleşen bir dalga yapısı olduğu olgusundan geliyordu. Dolayısıyla burada HMM oluştururken bağlı kalınan temel prensip 1:T aralığında gelen ivme serisindeki ivmelerin arasındaki geçişin bir anlam ifade edebileceği gerçeğine dayanmaktadır. Yani farklı hareketlerde, 1:T içerisindeki herhangi bir t anında gelecek olan ivmenin bir sınır aralığında olacağı ve bu ivme değerinden sonra gelecek t+1 anındaki bir sonraki ivme değerinin de belirli başka bir sınır aralığındaki ivme değerlerinden geleceği öngörüsünde bulunulmuştur. Ayrıca t ve t+1 anındaki bu iki sıralı ivmenin bulunduğu sınırların (aslında grup yani binler) bir birleri arasında da belirli bir olasılıksal geçiş durumu olduğu varsayımı yapılmıştır. Sonuç olarak her bir ivmenin belirli bir binden geldiği ve bu binlerin aralarında farklı hareketler için farklı olasılıksal oranda geçişler yaşanacağı varsayımı üzerinde durulmuştur. Bu sebeple her bir grup (her eksen için 10 tane), HMM model tabanlı bir sistem için bir durum (state) olarak kabul edilmiştir.

HMM yapıları için de Gruplanmış Dağılımla aynı sınır değerleri kullanılmıştır. Bu durum değerlerinin sınır değerleri Tablo 3.3'te gösterilmiştir. Saklı markov durumları belirlenirken esasında WISDM veri setinin yapısı incelenmiştir. WISDM veri setindeki veriler hareket türünden bağımsız bir şekilde -20,5 ile 20,5 m/s<sup>2</sup> aralığında bir dağılım göstermektedir. Gruplanmış dağılım değerlerinin ivmeleri 2,5 m/s<sup>2</sup> olarak bölmesinin temel sebebi dengeli bir dağılım olmasından kaynaklanmaktadır. Bu şekilde ivme değerleri bölündüğünde bir grup içerisine düşen ivme değerlerinin ortalaması ve bu ortalamadan sapma miktarları birbirlerine yakın olmaktadır. Her bir grup içerisindeki ivme değerlerini esasında benzer ortalama ve standart sapmaya sahip Gauss dağılımları olarak ifade edilmesi sağlanmaktadır. Böylelikle kurulacak modelde kullanılacak dağılımlar dengeli bir yapıda oluşturulabilir. Bu yüzden Saklı Markov modellerinde de durum değerleride bu aralıklarda belirlenmiştir.



Tablo 3.3. Saklı Markov modeli durumları

Durum Adı	Sınır Değeri (m/s <sup>2</sup> )
State#1	(-∞ ve -2,5)
State#2	[-2,5 ve 0)
State#3	[0 ve 2,5)
State#4	[2,5 ve 5)
State#5	[5 ve 7,5)
State#6	[7,5 ve 10)
State#7	[10 ve 12,5)
State#8	[12,5 ve 15)
State#9	[15 ve 17,5)
State#10	[17,5 ve +∞)

Geçiş olasılıkları hesaplanırken, kullanılan veri seti için sırayla 10 saniyelik veri okumaları (20Hz frekansında 50ms periyota sahip bu ivme serileri 200 adet veri okuması) gerçekleştirilmiştir. Her eksendeki 200 adet okumanın her t ve t+1 olmak üzere iki ardışık zaman verisinin bulunduğu durumlar belirlenmiştir. Bu durumlar arasındaki geçiş olasılıkları ise buna bağlı olarak hesaplanmıştır.

Geçiş olasılıklarının hesaplanması Şekil 3.1'de gösterilmiştir.

```

1 calculateTransitionsOfAnAxis(totalRecordsInAnAxis[M][N])
2   transitionMatrix[][] = {0}
3   totalTransition = 0
4   FOR i = 1 TO M
5     aRecord[] = totalRecordsInAnAxis[i];
6     FOR j = 1 to N
7       firstBin = findBinOfAcceleration(aRecord[j])
8       secondBin = findBinOfAcceleration(aRecord[j+1])
9       transitionMatrix[firstBin][secondBin] += 1
10    END FOR
11    totalTransition += (aRecord.length()-1)
12  END FOR
13  FOR i = 1 TO columnNumberOfTransitionMatrix
14    FOR j = 1 TO rowNumberOfTransitionMatrix
15      transitionMatrix[i][j] /= totalTransition
16    END FOR
17  END FOR
18 RETURN transtionMatrix

```

Şekil 3.1. Geçiş olasılıklarının hesaplanması

Kaba kodda bulunan ana fonksiyon MxN boyutunda bir eksendeki ivme değerlerini (TopRecordsInAnAxis) fonksiyon girdisi olarak almaktadır. Bu matris bir hareket türü için (Yürüme, Koşma, Oturma gibi herhangi bir insan aktivitesi) veri setindeki sadece bir eksene ait ivme serilerini tutmaktadır. Matristeki M ile ifade edilen satır sayısı esasında veri setindeki 1:T aralığındaki ivme serisini ifade etmektedir. N sütun sayısı ise o ivme serisindeki, ardışık ivmelerin indisini göstermektedir. Daha önce bahsedildiği üzere 200 adet kayıt okuduğumuzdan kabaca N sayısını 200 alabilir.

Daha formal olarak bu ifade Eşitlik 3.1’de gösterilmiştir.

$$\text{totalRecordsInAnAxis}[i] = \left[ O_{1:T}^k \right]^i \quad (3.1)$$

Bu ivme serileri alındıktan sonra, geçiş matrisi (transitionMatrix) ve toplam geçiş sayısı (total transition) başlangıç değerleri olarak 0 alınmıştır (2. ve 3. satırlar). Daha sonra her bir ivme serisine ait her ardışık iki ivme alınarak, ardışık ivmelerin düştükleri gruplar (bin) hesaplanır (7. ve 8. satırlar). Burada findBinOfAcceleration adlı metot ilgili ivme değerinin hangi durum aralığına (state) düştüğünü göstermektedir. Aslında ilgili fonksiyon gelen ivme değerini Tablo 3.3’teki durum aralıklarından birine eşleyen bir fonksiyondur.

Bu aşamada, geçiş matrisinde iki matris arasında bir geçiş olduğu bilgisi kayıt edilir. Bu arada toplam geçiş sayısı da birer birer arttırılmaktadır (9. satır).

Bu işlemden sonra geçiş matrisi değişkeni içerisinde her iki durum (state) arasındaki toplam geçiş sayısı tutulmaktadır. Fakat bunları HMM içerisinde geçiş sayısı olarak değil, geçiş olasılıkları olarak ifade edilmesi gerekmektedir. Bu sebeple geçiş matrisleri içerisindeki her satır-sütun ikilisi içerisindeki toplam geçiş sayısı, tüm sistemdeki toplam geçiş sayısına bölünerek olasılık değerlerine dönüştürülmektedir (13-17. satırlar arası). Hesaplanan geçiş olasılıkları fonksiyon geri dönüş değeri olarak bir üst fonksiyona döndürülmektedir.

Dikkat edilirse tasarlanan HMM’ler için herhangi iki durum arasında bir geçiş olasılığı olabilir. Bu türden bir HMM yapısı, Ergodik HMM (Ergodic HMM) olarak adlandırılmaktadır.

Bir HMM için geçiş olasılıkları gibi bir başka önemli parametrede başlangıç olasılıklarıdır. Yani bir Saklı Markov Modelinin ilk olarak zincirleme olasılık için, herhangi bir durumdan başlama olasılıklarının hesaplanması gereklidir. Bu işleme ait kaba kod Şekil 3.2’de verilmiştir.

```
1 calculateInitialProbsOfAnAxis(totalRecordsInAnAxis[M][N])
2   initalProbsOfAnAxis [] = {0}
3   FOR i = 1 TO M
4     initialAcceleration = totalRecordsInAnAxis[i][0];
5     binOfFirstAcc = findBinOfAcceleration(initialAcceleration);
6     initalProbsOfAnAxis[binOfFirstAcc] += 1 / M
7   END FOR
8 RETURN initalProbsOfAnAxis
```

Şekil 3.2. Başlangıç olasılıklarının hesaplanması

Başlangıç olasılıkları hesaplanırken geçiş olasılıkları ile benzer olarak bir eksen için ivme serilerinin tutulduğu aynı  $M \times N$  boyutundaki matris fonksiyon girdisi olarak alınmaktadır (1. satır). Başlangıç olasılıkları ilk olarak, tüm durumlar için 0 değeri belirlenir. 4 ve 6. satırlar arasında bir ivme serisine ait  $N$  boyutlu bir kaydın sadece ilk ivme değeri alınır ve bu değer karşılık geldiği durum aralığı (bin) tespit edilir. Ve bu durum değerini indis olarak alan bir matris içerisinde tutulur. 6. Satırda dikkat edilirse her bir durumla ilgili başlangıç olasılığı varlığı tespit edildiğinde, o durumdan başlama olasılığı  $1/M$  oranında arttırılmıştır. Bu işlemin sonucunda bir hareket türüne ait herhangi bir eksendeki başlangıç olasılıkları tespit edilmiş olunur. Fonksiyon geri dönüş değeri olarak bu başlangıç olasılıklarının tutulduğu matrisi bir üst fonksiyona aktarmaktadır.

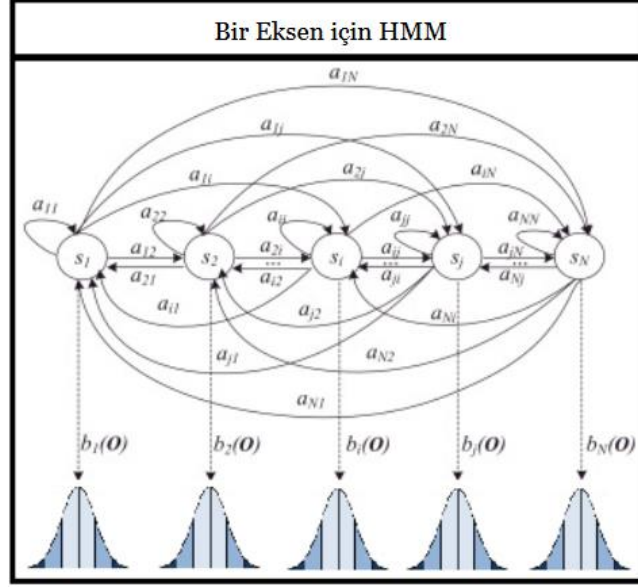
Emisyon olasılıkları ise her bir duruma düşen ivme değerlerinin bir Gauss dağılımı olarak gösterilmesidir. Burada Tablo 3.3'te yer alan durumlara ait her bir aralık değerindeki ivme değerlerinin ortalaması ( $\mu$ ) ve standart sapması ( $\sigma$ ) tespit edilir. Bir duruma ait geçiş olasılığı, o duruma ait ivme değerleri kullanılarak oluşturulmuş olan bir  $N(\mu, \sigma)$  parametrelili Gauss dağılımı ile ifade edilir.

Elde edilen saklı markov modelleri verisetindeki 200 boyutlu gözlem serilerinin tamamına uygulanarak olabirlik değerleri tespit edilmiştir. Bu işlemin sonucunda olarak 2.2.2'de açıklanan öznitelikler elde edilmiştir. Bu öznitelikler Tablo 3.4'te gösterilmiştir.

Tablo 3.4. Elde edilen öznitelikler

Ortalama ivme değeri (3 öznitelik)	Her eksen için ortalama ivme
Standart Sapma (3 öznitelik)	Her eksen için standart sapma
Ortalama toplam ivme (1 öznitelik)	$\sqrt{a_x^2 + a_y^2 + a_z^2}$
Ortalama mutlak sapma (3 öznitelik)	Her eksen için, gözlem serisinde bulunan ivme değerlerinin ilgili ekseninde ortalama ivmeden toplam sapma miktarı
HMM Öznitelikleri (18 öznitelik)	Gözlem serisine ait 3-eksende farklı hareket türlerinden gelme olasılıkları
Sınıf etiketi	Hareket türü

Elde edilen farklı hareket türlerine için her bir ekseninde oluşturulacak HMM'lere ait yapı Şekil 3.3'te gösterilmiştir. Dikkat edilmesi gereken nokta, saklı markov modelleri için ileriye yönelik olasılık algoritması kullanılarak hesap edilen olabirlik değerleri çok küçük değerler almaktadır. 200 boyutlu veri bir saklı markov modele giriş olarak uygulandığında aslında 199 adet geçiş olmaktadır.



Şekil 3.3. Bir eksen için HMM yapısı

200 adet geçiş sebebi ile elde edilen olasılık değerleri oldukça küçüktür. Dolayısıyla, saklı markov model geliştirilmesi süresinde bu yüksek hassasiyet değerlerine dayalı (high precision) Apfloat kütüphanesi kullanılmıştır. Apfloat kütüphanesi çok küçük veya çok büyük sayılarla çalışmasına imkan vererek, yüksek performansta ve standart programlama veri tiplerinden farklı olarak yüksek hassasiyette çalışmasına imkan verilmektedir [39].

### 3.2. Gerçekleştirilen Deneyler

3.1.2 ve 3.1.3'te açıklanan veri setleri oluşturulduktan sonra problem standart bir veri madenciliğindeki sınıflandırma problemi olarak ele alınmıştır. İki veri seti üzerinde de farklı sınıflandırma algoritmaları denenerek başarımlar karşılaştırılmıştır. Ayrıca iki öznelik uzayını incelemek için iki veri setinin sonuçları diğer veri setinin sonuçları ile karşılaştırılarak incelenmiştir.

Gerçekleştirilen testler WEKA makine öğrenmesi aracı ile gerçekleştirilmiştir. WEKA (Waikato Environment for Knowledge Analysis), veri madenciliği görevleri için bir makine öğrenme algoritmaları topluluğudur. Veri hazırlama, sınıflandırma, regresyon, kümeleme, birleşme kuralları madenciliği ve görselleştirme için araçlar içermektedir [40]. Java programlama dilinde yazılmış, hem kullanıcı arayüzüne hemde Java programlama dili için bir uygulama arayüzü (Application Interface – API) sunmaktadır.

Katışıklık Matrisinde (Confusion Matrix) düzenli bir görünüm sağlanması amacıyla hareket isimleri kısaltılmıştır. Bu kısaltmalara ilişkin detaylar Tablo 3.5'te verilmiştir.

Tablo 3.5. İnsan aktiviteleri için kısaltma tablosu

Kısaltma	Açıklama
JG	Koşma (Jogging)
WLK	Yürüme (Walking)
DW	Merdiven İnme (Downstairs)
UP	Merdiven Çıkma (Upstairs)
STD	Ayakta Durma (Standing)
SIT	Oturma (Sitting)
Has.	Hassasiyet
Kes.	Kesinlik
Doğ.	Doğ.
Top	Toplam Başarım

Aşağıda iki farklı veri seti üzerinde gerçekleştirilen sınıflandırma algoritmaları sonucu kurulan modelleri başarımları gösterilmiştir.

### 3.2.1. Sürekli zamanlı öznitelikler üzerinde gerçekleştirilen deneyler

Sürekli zamanlı öznitelikler ile oluşturulan 4617 satırlık veri seti üzerinden, toplam 6 farklı algoritma ile 10-Çapraz Geçerleme gerçekleştirilerek sınıflandırma modelleri kurulmuştur. Kurulan modellere ait katışıklık matrisleri (confusion matirx) Tablolar 3.6–3.11 arasında gösterilmiştir.

Tablo 3.6. C4.5 Karar Ağacı

	JG	WLK	DW	UP	STD	SIT	Kes.	Has.	Doğ.
JG	1165	8	5	14	0	0	0,966	0,977	%97,70
WLK	5	1871	47	65	0	0	0,926	0,941	%94,10
DW	10	61	269	66	0	1	0,699	0,661	%66,10
UP	26	81	62	354	2	1	0,704	0,673	%67,30
STD	0	0	2	1	220	3	0,969	0,973	%97,30
SIT	0	0	0	3	5	268	0,982	0,971	%97,10
Top:									%89,86

Tablo 3.7. Rastgele Ağaçlar (Random Forest – RF)

	JG	WLK	DW	UP	STD	SIT	Kes.	Has.	Doğ.
JG	1181	9	1	1	0	0	0,972	0,991	%99,10
WLK	0	1974	2	12	0	0	0,937	0,993	%99,30
DW	4	62	319	22	0	0	0,935	0,784	%78,40
UP	30	61	19	416	0	0	0,918	0,791	%79,10
STD	0	0	0	0	225	1	0,991	0,996	%99,60
SIT	0	0	0	2	2	272	0,996	0,986	%98,60
Top:									%95,06

Tablo 3.8. SVM-Çoklu Çekirdek (Poly Kernel)

	JG	WLK	DW	UP	STD	SIT	Kes.	Has.	Doğ.
JG	1174	18	0	0	0	0	0,946	0,985	%98,50
WLK	0	1962	8	18	0	0	0,813	0,987	%98,70
DW	21	183	154	49	0	0	0,815	0,378	%37,80
UP	46	249	27	204	0	0	0,747	0,388	%38,80
STD	0	0	0	2	221	3	0,991	0,978	%97,80
SIT	0	1	0	0	2	273	0,989	0,989	%98,90
Top:									%86,41

Tablo 3.9. SVM-RBF Çekirdeği

	JG	WLK	DW	UP	STD	SIT	Kes.	Has.	Doğ.
JG	1107	85	0	0	0	0	0,945	0,929	%92,90
WLK	15	1973	0	0	0	0	0,662	0,992	%99,20
DW	6	401	0	0	0	0	N/A	0	%00,00
UP	44	482	0	0	0	0	N/A	0	%00,00
STD	0	37	0	0	187	2	0,930	0,827	%82,70
SIT	0	2	0	0	14	260	0,992	0,942	%94,20
Top:									%76,43

Tablo 3.10. Çok Katmanlı Algılayıcı (Multilayer Perceptron – MLP)

	JG	WLK	DW	UP	STD	SIT	Kes.	Has.	Doğ.
JG	1170	9	8	5	0	0	0,970	0,982	%98,20
WLK	1	1922	22	43	0	0	0,947	0,967	%96,70
DW	6	36	322	43	0	0	0,785	0,791	%79,10
UP	29	61	58	377	0	1	0,802	0,717	%71,70
STD	0	0	0	2	223	1	0,991	0,987	%98,70
SIT	0	1	0	0	2	273	0,993	0,989	%98,90
Top:									%92,90

Tablo 3.11. K-NN (1-NN)

	JG	WLK	DW	UP	STD	SIT	Kes.	Has.	Doğ.
JG	1189	0	1	2	0	0	0,992	0,997	%99,70
WLK	0	1970	6	12	0	0	0,985	0,991	%99,10
DW	2	13	370	22	0	0	0,907	0,909	%90,90
UP	7	16	29	474	0	0	0,929	0,901	%90,10
STD	0	0	0	0	224	2	0,996	0,991	%99,10
SIT	0	0	2	0	1	273	0,993	0,989	%98,90
Top:									%97,51

Tablolar incelendiğinde, sürekli zamanlı öznitelikler için en başarılı modellerin 1-NN modelidir. Oluşturulan diğer modeller ile bu model arasındaki başarımların temel sebebi Merdiven inme ve çıkma (DW ve UP) hareketlerinden kaynaklanmaktadır. Tüm modellerde en düşük sınıflandırma başarımlarına sahip olan ve dolayısıyla diğer tüm sınıflarla katışıklık oranı en yüksek olan sınıflar DW ve UP sınıflarıdır.

Bu problemin temel sebebi DW ve UP hareketlerinde, sınıflar arası benzerliğin (inter-class similarity) fazla olmasıdır. Diğer hareketlerin aksine DW ve UP hareketleri aslında bir çeşit yürüme hareketleridir. Dolayısıyla hem kendi aralarında hem de yürüme hareketi ile benzerlikleri oldukça fazladır. Bu durum yukarıdaki tablolara bakıldığında açıkça anlaşılabilir. DW hareketi en çok yürüme (WLK) daha sonra ise merdiven çıkma (UP) hareketi ile karıştırılmıştır. Aynı durum UP içinde görülmüştür. UP için DW ile benzer şekilde önce WLK daha sonra DW ile karıştırılmıştır.

K-NN algoritması, 1 komşu için gözle görülür biçimde daha iyi bir performans vermiştir. Bu modelin diğer yöntemlerden en önemli farkı aslında sadece yakınlığa bakarak sınıflandırma yapmasıdır. Dolayısıyla standart bir sınıflandırma modelinden farklı olarak en yakın komşunun etiketi ile gelen örneği sınıflandırmaktadır. Bu gözleme dayanarak sınıflar arası katışıklık olsa da, sınıf içi örneklerin birbirlerine yakın olduğu yorumu yapılabilir. Bir diğer başarılı model olan RF algoritmasına kıyasla, DW için %12,5'lik, UP için ise %11'lik bir başarı artışı söz konusudur. Yani iki sınıf için başarımlar gözle görülür bir biçimde değişmiştir. Genel başarımlar ise %97.51'dir.

### 3.2.2. HMM öznitelikleri üzerinde gerçekleştirilen deneyler

HMM özniteliklerinin bulunduğu, 4615 satırlık veri seti üzerinde 6 farklı sınıflandırma algoritması ve 10-Çapraz Geçerleme işlemi kullanılarak toplamda 6 adet model oluşturulmuştur. Bu modellere ait sonuçlar Tablolar 3.12–3.17 arasında gösterilmiştir.

Tablo 3.12. C4.5 Karar Ağacı

	JG	WLK	DW	UP	STD	SIT	Kes.	Has.	Doğ.
JG	1162	8	7	16	0	0	0,966	0,974	%97,40
WLK	2	1858	54	75	0	0	0,920	0,934	%93,40
DW	10	68	275	54	0	0	0,694	0,676	%67,60
UP	29	83	58	353	2	1	0,706	0,671	%67,10
STD	0	2	1	0	219	4	0,978	0,969	%96,90
SIT	0	0	1	2	3	270	0,982	0,978	%97,80
Top:									%89,60

Tablo 3.13. Rastgele Ağaçlar

	JG	WLK	DW	UP	STD	SIT	Kes.	Hassasiyet	Doğ.
JG	1181	8	3	1	0	0	0,982	0,990	%99,00
WLK	0	1969	2	18	0	0	0,938	0,990	%99,00
DW	2	54	319	32	0	0	0,938	0,784	%78,40
UP	20	68	16	422	0	0	0,888	0,802	%80,20
STD	0	0	0	0	226	0	0,991	1,000	%100,00
SIT	0	0	0	2	2	272	1,000	0,986	%98,60
Top:									%95,06

Tablo 3.14. SVM- Çoklu Çekirdek

	JG	WLK	DW	UP	STD	SIT	Kes.	Has.	Doğ.
JG	1167	26	0	0	0	0	0,960	0,978	%97,80
WLK	1	1981	0	7	0	0	0,745	0,996	%99,60
DW	11	317	3	76	0	0	1,000	0,007	%00,70
UP	37	334	0	152	3	0	0,639	0,289	%28,90
STD	0	0	0	2	222	2	0,992	0,953	%95,30
SIT	0	0	0	1	12	263	0,835	0,820	%82,00
Top:									%82,04

Tablo 3.15. SVM-RBF Çekirdeği

	JG	WLK	DW	UP	STD	SIT	Kes.	Has.	Doğ.
JG	1156	37	0	0	0	0	0,915	0,969	%96,90
WLK	69	1917	0	0	3	0	0,672	0,964	%96,40
DW	6	401	0	0	0	0	N/A	0	%0,00
UP	32	491	0	0	3	0	N/A	0	%0,00
STD	0	5	0	0	221	0	0,880	0,978	%97,8
SIT	0	2	0	0	24	250	1,000	0,906	%90,6
Top:									%76,75

Tablo 3.16. MLP

	JG	WLK	DW	UP	STD	SIT	Kes.	Has.	Doğ.
JG	1163	14	7	9	0	0	0,973	0,975	%97,50
WLK	6	1903	33	45	1	1	0,935	0,957	%95,70
DW	4	44	298	58	0	3	0,723	0,732	%73,20
UP	22	72	73	358	1	0	0,757	0,681	%68,10
STD	0	0	0	2	222	2	0,987	0,982	%98,20
SIT	0	2	1	1	1	271	0,978	0,982	%98,20
Top:									%91,30

Tablo 3.17. 1-NN

	JG	WLK	DW	UP	STD	SIT	Kes.	Has.	Doğ.
JG	1184	3	2	4	0	0	0,988	0,992	%99,2
WLK	6	1953	16	20	0	0	0,949	0,982	%98,2
DW	3	41	324	39	0	0	0,876	0,796	%79,6
UP	11	59	28	428	0	0	0,868	0,814	%81,4
STD	0	1	0	0	225	0	0,996	0,996	%99,6
SIT	0	0	0	2	1	273	1,000	0,989	%98,9
Top:									%95,02

Sürekli zamanlı öznitelik uzayında karşılaşılan biçimde en başarılı modeller RF ve 1-NN olmuştur. Elde edilen sonuçlar sürekli zamanlı özniteliklerin kullanıldığı durumla benzerlik göstermektedir. DW ve UP hareketleri sınıflar arası benzerlik sebebi ile yürüme ve kendi aralarında karışmaktadır. Dolayısıyla buna bağlı olarak başarımlar bu sınıflarda düşük olmaktadır. En başarılı iki model incelendiğinde RF algoritmasının DW hareketi için %78,40, UP hareketi için %80,20'de kaldığını benzer şekilde 1-NN



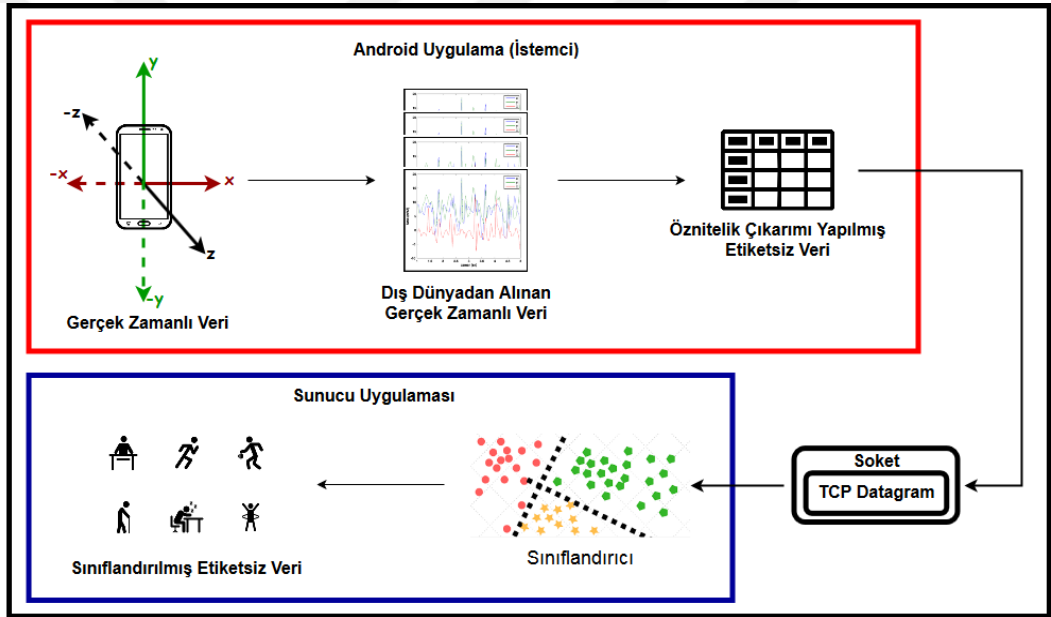
algoritması incelendiğinde başarı oranının DW hareketinde %79,6, UP hareketinde ise %81,4'te kaldığı gözlemlenmiştir.

İki farklı veri seti için en başarılı sonuçlar incelendiğinde, sınıflar arası benzerlik söz konusu olduğunda sürekli zamanlı özniteliklerin DW ve UP hareketlerini ayırmada net bir şekilde daha iyi olduğu gözlemlenmektedir. Dalga özniteliklerini kullanan veri seti için en başarılı model olan 1-NN (Tablo 3.11) ile HMM öznitelikleri kullanan veri setine ait en başarılı model olan RF (Tablo 3.13) incelendiğinde;

- DW ve UP hareketleri için sürekli zamanlı dalga öznitelikleri, HMM özniteliklerine oranla net bir biçimde daha iyi bir başarı sergilemektedir. Bu ise sürekli zamanlı dalga özniteliklerinin sınıflar arası benzeşim için daha etkin olduğunu göstermektedir.
- STD hareketi için HMM öznitelikleri daha iyi çalışmaktadır (yaklaşık %0,90)
- Diğer hareket türleri için az farkla sürekli zamanlı öznitelikler daha başarılıdır.

#### 4. GERÇEK ZAMANLI UYGULAMA ÖRNEĞİ

Kurulan sınıflandırma modelinin gerçek zamanlı gerçek bir senaryoda ne kadar etkili olduğunun tespit edilebilmesi amacıyla gerçek zamanlı bir istemci-sunucu mimarisi oluşturulmuştur. Gerçekleştirilen istemci-sunucu yapısındaki istemci uygulaması anlık olarak 50Hz frekansında ivmeölçer verisi toplayan bir akıllı telefon uygulamasıdır. Bu uygulama Samsung Note 2 telefonunda çalıştırılmıştır. Sistemin ikinci bileşeni ise, sınıflandırıcının yer aldığı sunucu uygulamasıdır. Bu iki aşamalı sistemin yapısı Şekil 4.1'de gösterilmiştir.



Şekil 4.1. İstemci-Sunucu Yapısı

İstemci uygulaması ön cebe yerleştirilmiş olan Android telefon üzerinde çalışmaktadır. 10 saniyelik periyotlarla toplamda 200'e yakın kayıt toplanarak öznitelik çıkarımı yapılmaktadır. Öznitelik çıkarımı yapılmış etiketsiz veri ise kablosuz bir ağ üzerinden sunucu uygulamasına gönderilmektedir. Sunucu uygulaması ise kendi içerisinde daha önce 2. Bölümde açıklanan ve 3. Bölümde deneysel olarak gerçekleştirilen işlemlerin uygulanarak elde edilen bir sınıflandırıcıya sahiptir. Sunucu uygulaması bir Android telefon üzerinden gelen ilgili veriyi alarak kendi içerisinde bulunan sınıflandırıcı ile etiketsiz veriyi sınıflandırır. Böylelikle ilgili hareket türüne ait bir tahmin yapılmaktadır. Bu uygulama toplam 5 farklı denekten toplanmıştır. Her denekten,

telefonu sağ ön ceplerine yerleştirilmeleri istenmiştir. 5 denekten de yürüme, koşma, merdiven inme, merdiven çıkma, ayakta durma ve oturma hareketlerini gerçekleştirmeleri istenmiştir. Yaklaşık 1 saatlik deney sonucunda toplam 70.600 veri noktası ve ayrıklaştırılmaları ile 353 adet farklı hareket örüntüsü gerçek zamanlı olarak sınıflandırılmıştır. Sürekli zamanlı öznitelikler kullanılarak, C4.5 algoritması ile eğitilen sınıflandırıcı (Tablo 3.6) bu sistem için kullanılmıştır. Elde edilen sonuçlar Tablo 4.1'de gösterilmiştir.

Tablo 4.1. Gerçek Zamanlı Uygulama için Katışıklık Matrisi

	JG	WLK	DW	UP	STD	SIT	Kes.	Hassasiyet	Doğ.
JG	53	0	5	1	0	0	0,929	0,898	%89,83
WLK	3	43	23	13	0	0	0,597	0,524	%52,43
DW	0	6	25	6	0	0	0,416	0,532	%53,19
UP	0	17	7	13	0	0	0,351	0,351	%35,13
STD	0	3	0	2	67	5	1,000	0,870	%87,01
SIT	1	3	0	2	0	58	0,921	0,906	%90,63
Top:									%73,71

Gerçek zamanlı uygulamada Koşma, Oturma ve Ayakta durma hareketleri için başarımlar iyi düzeyde olsada, merdiven inme ve çıkma ile yürüme hareketleri için başarımlar oldukça düşüktür. Bu sınıflar kendi aralarında yüksek katışıklığa sahiptir. Özellikle merdiven çıkma hareketi için başarımlar %50'nin altındadır. Dolayısıyla genel başarımlar gerçek zamanlı bir senaryoda %73,71'de kalmıştır.

## 5. SONUÇLAR VE ÖNERİLER

Bu tez çalışması kapsamında, 3-eksenli ivmeölçer verisi kullanılarak bir insan aktivite tespit sistemi tasarımı gerçekleştirilmiştir. İnsan aktivitesi tespiti için, WISDM veri seti kullanılmıştır. Bu ham veriden sürekli zamanlı dalga özniteliklerinin kullanıldığı bir veri seti elde edilmiştir. İkinci olarak HMM öznitelikleri olarak adlandırabileceğimiz, belirli bir aralıktaki bir hareket eyleminin hangi insan aktivitesine ne ölçüde benzediği bilgisinin kullanıldığı ikinci bir veri seti olmak üzere 2 farklı öznitelik uzayına ait veri setleri oluşturulmuştur.

Elde edilen iki farklı veri seti üzerinde farklı sınıflandırma algoritmaları çalıştırılmış ve başarımları hem veri setlerinin kendi içerisinde (kurulan farklı modeller ile) hem de diğer veri setiyle karşılaştırılması gerçekleştirilmiştir.

Yapılan deneyler sonucunda sürekli zamanlı özniteliklerin, Saklı Markov Modellerinin ileriye yönelik olasılık değerlerinin kullanıldığı özniteliklere kıyasla daha başarılı olduğu sonucuna varılmıştır. Sınıflar düzeyinde başarımlar, 4 farklı harekette çok büyük önem arz etmezken (koşma, yürüme, oturma ve ayakta durma), 2 farklı harekette (merdiven inme ve merdiven çıkma) gözle görülür bir biçimde fark edilmektedir.

Bölüm 3'te bulunan, iki veri setinin en başarılı modelleri kendi aralarında kıyaslandığında, sürekli zamanlı öznitelik kullanan veri seti kullanılarak oluşturulan k-NN modeli, HMM öznitelikleri kullanan Rastgele Ağaç algoritmasından, merdiven inme için %12,50, merdiven çıkma hareketinde ise %9,10 daha başarılıdır. Bu sonuçta bakarak net bir biçimde sürekli zamanlı öznitelik uzayının HMM çıktılarının öznitelik olarak kullanılmasından daha başarılı olduğu sonucu çıkarılabilir.

Bölüm 3'te gerçekleştirilen deneylerde, merdiven inme ve merdiven çıkma hareketleri kendi içerisinde incelendiğinde en başarılı modeller de dahi diğer hareketlere oranla gözle görülür biçimde daha az başarımla sahip olduğu gözlemlenmektedir. Bu hareketlerin özellikle yürüme hareketi ile karıştırıldığı gözlemlenmektedir. Yürüme hareketinden sonra ise merdiven inme genellikle merdiven çıkma, merdiven çıkma hareketinin ise merdiven inme hareketi ile karıştırıldığı gözlemlenmiştir. Bu durum Bölüm 4'te gerçekleştirilen gerçek bir uygulamada benzerlik göstermekle birlikte, etkisi itibari ile eğitim seti üzerindeki sonuçlardan daha açık bir biçimde ortaya

çıkıştır. Gerçek zamanlı uygulamada merdiven inme ve çıkma hareketlerine ilave olarak yürüme hareketinde de diğer hareket türleri ile yüksek katışıklık oranı gözlemlenmiştir. Elde edilen sonuçlardan hareketle, bu hareket türleri için sınıflar arası benzerlik oranının (inter-class similarity) oldukça yüksek olduğu sonucuna varılabilir.

Ayrıca yapılan deneyler sonucunda en başarılı sonuçların 1-NN ve Rastgele Ağaç gibi ağaç algoritmalarının en başarılı modeller olmaları dikkat çekicidir. İki farklı öznitelik uzayında da en başarılı iki model 1-NN ve Rastgele Ağaç olmuştur. 1-NN algoritmasının bu denli başarılı olmasını sebeplerinden biride veri setinin genelinde sınıflar arası veri noktalarının birbirlerine yakınlığıdır. Dolayısıyla en yakın komşuya bakıldığında genellikle bu bizi doğru sonuca götürmektedir. Sınıflar arası karışım özellikle iki sınıf özelinde fazla da olsa tüm sınıflarda sınıf örneklerinin birbirlerine yakınlığının bir anlam ifade ettiği açıkça görülmektedir. Bu iki sınıflandırma algoritması insan aktiviteleri tespit sistemlerinde çok fazla kullanılmamıştır. Bu algoritmaların kullanılmadığı varsayıldığında MLP ve C4.5 algoritmaları en başarılı modellerdir. Bu modeller literatürde sıklıkla kullanılmakta olup genellikle insan aktiviteleri tespiti ile ilgili literatürde başarılı algoritmalar olarak yer almaktadır (Bölüm 1.3.1). Dolayısıyla elde edilen sonuçlar literatürdeki sonuçları destekler niteliktedir.

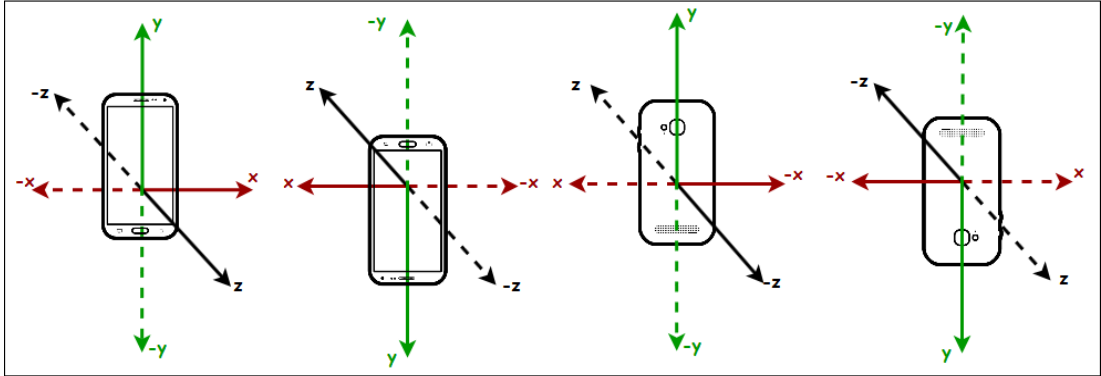
Bu etkiyi azaltmak adına [32] ve [3, 34]'te yapılan çalışmalara benzer şekilde iki aşamalı modeller kurulabilir. Öncelikli olarak Merdiven İnme ve Çıkma hareketleri birleştirilerek ortak bir hareket olarak kabul edilip, 5 sınıflı bir aktivite tespiti gerçekleştirilebilir. Bu aşamadan sonra ise bu birleştirilmiş hareket türü olarak tespit edilmiş veri noktaları, sadece bu iki sınıf kullanılarak eğitilmiş olan ikinci bir model aracılığıyla sınıflandırılabilir. 6 sınıf arasında bir ayrılabilirlik problemi çözmektense, 2 sınıf arasındaki ayrılabilirliğin tespiti için bir modelin kurulması daha basittir ve bu başarıyı arttırabilir.

Esasında kullanılan veri seti, sadece ivmeölçer verilerinden oluştuğu için öznitelik olarak kullanılacak veriler sadece ivmeölçer verisinden çıkartılmaktadır. Merdiven inme ve çıkma hareketleri çoğu açıdan birbirlerine benzeseler dahi yön bilgisi açısından farklılık göstermektedir. Örneğin jiroskop verisinin kullanılması ile yön ölçümünün hassas bir şekilde gerçekleştirilmesi bu iki hareket türünün birbirlerinden ve yürüme hareketinden ayırt edilmesi için bir önem arz edebilir.

Jiroskop verisi WISDM veri seti gibi veri setinde bulunmasa dahi, jiroskop dışında esasında farklı eksenlerdeki ivme değerlerinin birbirleri ile aralarındaki açı bilgilerinin elde edilmesi veya bileşke vektörler üzerinden bir öznitelik çıkarımı bu sınıflar arası benzerliğin azaltılması yönünde önemli bir katkı sağlayabilir. Gelecek bir çalışma olarak bu türden özniteliklerin çıkarılması için bir çalışma gerçekleştirilmesi planlanmaktadır.

Gerçekleştirilen deneyler ve elde edilen sonuçlar WISDM veri setine göre gerçekleştirilmiştir. Bu veri setinin laboratuvar ortamında toplanmıştır. Dolayısıyla gerçekleştirilen insan aktivitelerinin gerçek bir kullanım senaryosu içerisinde elde edilmediği bilinmelidir. Kullanılan verinin toplanmasında bazı önemli kısıtlamalar mevcuttur. Bu kısıtlamalardan ilki insan aktivitelerinin düzenli bir biçimde gerçekleştirildiği gerçeğidir. Yani esasında farklı denekler bu aktiviteleri belirli sınırlar çerçevesinde düzenli bir biçimde gerçekleştirmektedir. Bu sebeple gerçek dünya verisini tam anlamıyla yansıtması mümkün değildir. Bu sebeple bu veri setinin, gerçekleştirdiğimiz gerçek zamanlı bir senaryoyu tam olarak yansıtması son derece doğaldır. Böyle bir durum için daha farklı bir veri toplama aşaması gerekebilir.

Önemli bir diğer kısıtlama ise bir mobil telefonun deneklerin cebindeki yerleştirilme biçimidir. Bu durum Şekil 5.1'de gösterilmiştir.



Şekil 5.1. Bir mobil telefonun alabileceği farklı şekiller

Telefonun alacağı farklı yerleştirme biçimlerine bağlı olarak aynı denek aynı aktiviteyi gerçekleştirirse dahi etki edeceği eksenler farklı olabilmektedir. Bu sebepten dolayı gerçek zamanlı bir uygulama için sadece mobil cihazın tek bir şekilde yerleştirilerek toplanmış olan veriler gerçek bir uygulama için uygun değildir. WISDM veri seti Şekil 5.1'deki ilk duruma benzer bir şekilde ön sağ cebe yerleştirilerek toplanmıştır. Bölüm 4'te gerçek uygulamada telefon yönelimini rasgele bir şekilde alarak deneylerimizi

gerçekleřtirdik. Bu sebeple Bölüm 3'teki ve Bölüm 4'teki sonuçları birbirinden bu kadar farklı çıkması beklenebilir. Gerçek zamanlı bir uygulamada kullanılacak olan uygun nitelikteki bir veri setinin toplanabilmesi için telefon yönelimi dikkate alınarak veri toplamalı veya bu durumun etkilerinin azaltılacağı yöntemlerin (örneğin yönelim bağımsız bir öznelik uzayının tespit edilmesi) izlenmesi gerekmektedir.

Uygulama ile alakalı bir diğerk durum ise sensörlerin hassasiyeti ile alakalıdır. Daha önce WISDM veri setinin 50Hz frekansında verilerden oluştuğundan bahsedilmiştir. Gerçekte durum buna yakın olsada neredeyse hiçbir zaman 50 Hz frekansında ayarlanmış olan bir ivmeölçer kusursuz bir biçimde 50 Hz frekansında veri toplayamamaktadır. Bir interpolasyon işlemi gerçekleştirilerek bu ivmeölçerin küçük sapmalarla hiçbir zaman 50 Hz olarak toplayamayacağı veri noktaları iyileştirilebilir. Burada bir interpolasyon işleminin başarımı arttıracığının garantisi verilmemektedir. Gelecek bir çalışma olarak bir interpolasyon işleminin insan aktivitelerine başarımını nasıl etkileyeceği deneysel olarak denenmesi ve bu konu ile alakalı bir yayın çıkarılması hedeflenmektedir.

## KAYNAKLAR

- [1] Statista, <https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/> (Ziyaret Tarihi: 31 Aralık 2018).
- [2] Cisco, VNI Global Fixed and Mobile Internet Traffic Forecasts, <https://www.cisco.com/c/en/us/solutions/service-provider/visual-networking-index-vni/index.html#~mobilevni> (Ziyaret Tarihi: 10 Mayıs 2018)
- [3] Ranao C., Cho S., Human Activity Recognition Using Smartphone Sensors With Two-Stage Continuous Hidden Markov Models, *International Conference on Natural Computation (ICNC)*, Xiamen, China, Aug. 19-21, 2014.
- [4] Lester J., Choudhury T., Borriello G., A Practical Approach to Recognizing Physical Activities, *International Conference on Pervasive Computing*, Ireland, May 07-10, 2006, 1-16.
- [5] Bayat A., Pomplun M., Tran D. A., A study on human activity recognition using accelerometer data from smartphones. *Procedia Computer Science*, **34**, 450-457.
- [6] Hsieh C.Y., Huang C.N., Liu K.C., Chu W.C., Chan C.T., A machine learning approach to fall detection algorithm using wearable sensor, *Advanced Materials for Science and Engineering (ICAMSE)*, Tainan, Taiwan, Nov. 12-13, 2016, 707-710.
- [7] Catalbas B., Yucesoy B., et al, A comparative study of classification methods for fall detection, *Signal Processing and Communications Applications Conference (SIU)*, Trabzon, Turkey, Apr. 23-25, 2014, 1315-1318.
- [8] Erdogan S.Z., Bilgin T.T., A data mining approach for fall detection by using k-nearest neighbor algorithm on wireless sensor network data, *IET Communucations*, 2012, **6**(18), 3281–3287.
- [9] Lee T., Mihailidis A., An intelligent emergency response system: preliminary development and testing of automated fall detection, *Journal of telemedicine and telecare*, 2005, **11**(4), 194-198.
- [10] Rougier C., Meunier J., St-Arnaud A., Rousseau J., Robust video surveillance for fall detection based on human shape deformation, *IEEE Transactions on circuits and systems for video Technology*, 2011, **21**(5), 611-622.
- [11] Kurban O.C., Yildirim T., Neural network based daily activity recognition without feature extraction, *Signal Processing and Communications Applications Conference (SIU)*, Apr. 23-25, 2014, 567-570.



- [12] Wei L., Shah S.K., Human Activity Recognition using Deep Neural Network with Contextual Information, *International Conference on Computer Vision Theory and Applications (VISAPP)*, 2017, 34-43.
- [13] Legendre, A. M., Nouvelles méthodes pour la détermination des orbites des comètes, *F. Didot*, 1805.
- [14] Bayes T., Price R., Canton J., An essay towards solving a problem in the doctrine of chances, 1763.
- [15] O'Connor J.J., Robertson E.F., Laplace P.S., <http://www-groups.dcs.st-and.ac.uk/history/Biographies/Laplace.html> (Son Ziyaret: 31 Aralık 2018).
- [16] Markov A.A., Example of statistical research on text of "Eugene Onegin", illustrating interconnection of trials in chain, *Izvestiya Akademii Nauk SPb*, 1913, **6**, 153-162.
- [17] Luckert M., Schafer-Kenhert M., Using Machine Learning Methods for Evaluating the Quality of Technical Documents, Master Thesis, Linnaeus University, Växjö, Sweden, 2016.
- [18] Quinlan J.R., *C4. 5: programs for machine learning*, 1st ed., Morgan Kaufmann Publishers, Massachusetts, ABD, 1993.
- [19] Quinlan, J.R., Induction of decision trees, *Machine learning*, 1986, **1**(1), 81-106.
- [20] Akin Ç.E., Random Forest Classification, [cagriemreakin.com/veri-bilimi/random-forest-classification-10.html](http://cagriemreakin.com/veri-bilimi/random-forest-classification-10.html) (Ziyaret Tarihi: 30 Aralık 2018).
- [21] Alpaydın E., *Yapay Öğrenme*, 4. Baskı, Boğaziçi Üniversitesi Yayınevi, İstanbul, 2018.
- [22] Honkela A., Hidden Markov Models, Continuous observations, <http://www.helsinki.fi/~ahonkela/dippa/node35.html> (Ziyaret Tarihi: 25 Aralık 2018).
- [23] Rabiner L.R., A tutorial on hidden Markov models and selected applications in speech recognition, *Proceedings of the IEEE*, 1989, **77**(2), 257-286.
- [24] Bao L., Intille S., Activity recognition from user-annotated acceleration data, *International Conference on Pervasive Computing*, Berlin, Germany, 2004, 1-17.
- [25] Banos O., Damas M., Pomares H., Prieto A., and Rojas I., Daily living activity recognition based on statistical feature quality group selection, *Expert Systems with Applications*, 2012, **39**(9), 8013–8021.
- [26] Ravi N., Dandekar N., Mysore P., Littman M.L., Activity recognition from accelerometer data, *Innovative applications of artificial intelligence*, Pittsburg, ABD, July 09, 2005, 1541-1546.

- [27] Kwapisz J.R., Weiss G.M., Moore S.A., Cell phone-based biometric identification, *Theory Applications and Systems (BTAS)*, Washington, DC, USA, Sept. 27-29, 2010, 1-7.
- [28] Kwapisz J. R., Weiss G. M., Moore S.A., Activity Recognition Using Cell Phone Accelerometers, *SIGKDD Explor. Newsl.*, New York, NY, USA, December, 2011, **12**(2), 74-82.
- [29] Chernbumroong S., Atkins A. S., and Yu H., Activity classification using a single wrist-worn accelerometer. In *Software, Knowledge Information, Industrial Management and Applications (SKIMA)*, 2011, 1-6.
- [30] Wu W., Dasgupta S., Ramirez E.E., Peterson C., Norman G.J., Classification accuracies of physical activities using smartphone motion sensors, *Journal of Medical Internet Research*, 2012, **14**(5), 5317-5337.
- [31] Anjum A., Ilyas M. U., Activity Recognition using smartphone sensors, *Consumer Communications and Networking Conference (CCNC)*, Las Vegas, NV, USA, Jan. 11-14, 2013, 914-919.
- [32] Lee Y., Cho S., Activity Recognition Using Hierarchical Hidden Markov Models on a Smartphone with 3D Accelerometer, *International Conference on Hybrid Artificial Intelligence Systems (HAIS)*, 2011, 460-467.
- [33] Kim Y., Kang B., Kim D., Hidden Markov Model Ensemble for Activity Recognition using Tri-axis Accelerometer, *IEEE International Conference on Systems, Man and Cybernetics (SMC2015)*, Kowloon, China, Oct. 9-12, 2015, 3036-3041.
- [34] Ranao C., Cho S., Recognizing Human Activities from Smartphone Sensors Using Hierarchical Continuous Hidden Markov Models, *International Journal of Distributed Sensor Networks*, 2017, **13**, 1-16.
- [35] Han C.W., Kang S.J., Kim N.S., Implementation of hmm-based human activity recognition using single triaxial accelerometer. *IEICE transactions on fundamentals of electronics, communications and computer sciences*, 2010, **93**(7), 1379-1383.
- [36] Mannini A., Sabatini A.M., Machine learning methods for classifying human physical activity from on-body accelerometers, *Sensors*, Feb 1, 2010, 10(2), 1154-1175.
- [37] Weiss G.M., Hirsh H., Learning to predict rare events in event sequences, *International Conference on Knowledge Discovery and Data Mining*, 1998, AAAI Press, Menlo Park, CA, 359-363.
- [38] Preece S.J., Goulermas J.Y., Kenney L., Howard D., A Comparison of Feature Extraction Methods for the Classification of Dynamic Activities From Accelerometer Data, *IEEE Transactions on Biomedical Engineering*, 2009, **56**(3), 871-879.

[39] Apfloat for Java, Mikko Tommila, [http://www.apfloat.org/apfloat\\_java/](http://www.apfloat.org/apfloat_java/) (Ziyaret Tarihi: 10 Ocak 2019).

[40] WEKA Software in Java, <https://www.cs.waikato.ac.nz/ml/weka/> (Ziyaret Tarihi: 10 Ocak 2019).



## KİŞİSEL YAYIN VE ESERLER

- [1] **Eren K.K.**, Küçük K., Machine learning based real-time activity detection system design, *International Conference on Computer Science and Engineering (UBMK)*, Antalya, Turkey, October 5-8, 2017.



## ÖZGEÇMİŞ

Kazım Kıvanç Eren, 1994 yılında İstanbul'da doğdu. 2012 yılında girdiği Kocaeli Üniversitesi Bilgisayar Mühendisliği bölümünden 2016 yılında mezun olmuştur. 2019 yılında, Kocaeli Üniversitesi Fen Bilimlerin Enstitüsüne bağlı Bilgisayar Mühendisliği Anabilim Dalı'nda yüksek lisans eğitimini tamamlamıştır. Yüksek lisans eğitimi sırasında, "Machine learning based real-time activity detection system design" başlıklı bir çalışma yayınlamıştır. 2018 yılından itibaren özel bir Ar-Ge merkezinde Ar-Ge Mühendisi olarak "Teknoloji ve Yenilik Destek Programları Başkanlığı" (TEYDEB) tarafından desteklenen çeşitli Ar-Ge projelerinde yer almaktadır.

