



T.C.
İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



YÜKSEK LİSANS TEZİ

SAĞLIK VERİLERİNİN ANALİZİ İÇİN MOBİL VE BULUT
YAZILIMI TASARIMI

Gökhan ÖZOĞUR

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

DANIŞMAN

Dr. Öğr. Üyesi Muhammed Ali AYDIN

Haziran, 2018

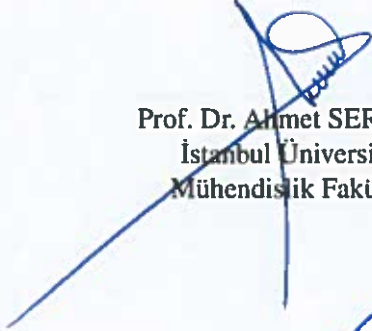
İSTANBUL

Bu çalışma 06.06.2018 tarihinde aşağıdaki jüri tarafından Bilgisayar Mühendisliği Anabilim Dalı Bilgisayar Mühendisliği Programında Yüksek Lisans Tezi olarak kabul edilmiştir.

Tez Jürisi



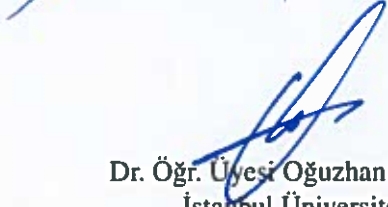
Dr. Öğr. Üyesi Muhammed Ali AYDIN (Danışman)
İstanbul Üniversitesi
Mühendislik Fakültesi



Prof. Dr. Ahmet SERTBAŞ
İstanbul Üniversitesi
Mühendislik Fakültesi



Doç. Dr. Yasin ÖZÇELEP
İstanbul Üniversitesi
Mühendislik Fakültesi



Dr. Öğr. Üyesi Oğuzhan ÖZTAŞ
İstanbul Üniversitesi
Mühendislik Fakültesi



Dr. Öğr. Üyesi Şenay BAHİYYAR
İstanbul Teknik Üniversitesi
Bilgisayar ve Bilişim Mühendisliği
Fakültesi



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

ÖNSÖZ

Yüksek lisans eğitimim ve tez çalışmam süresince bilgi, birikim ve tecrübeleri ile bana yol gösteren ve yardımlarını esirgemeyen değerli tez danışmanım Dr. Öğr. Üyesi Muhammed Ali AYDIN'a saygı ve şükranlarımı sunarım.

Ayrıca tüm hayatım boyunca yanımda olan ve eğitimim için maddi ve manevi desteklerini esirgemeyen sevgili aileme teşekkürlerimi bir borç bilirim.

Haziran, 2018

Gökhan ÖZOĞUR



İÇİNDEKİLER

	Sayfa No
ÖNSÖZ	iv
İÇİNDEKİLER	vi
ŞEKİL LİSTESİ	vii
TABLO LİSTESİ	viii
SİMGE VE KISALTMA LİSTESİ	ix
ÖZET	x
SUMMARY	xi
1. GİRİŞ	1
2. GENEL KISIMLAR	4
2.1. MOBİL SAĞLIK	4
2.1.1. Mobil Sağlık Alanında Yapılmış Çalışmalar	4
2.1.2. Sağlık Verileri	5
2.2. VERİ ANALİZİ	6
2.2.1. Doğrusal Regresyon	6
2.2.2. Yapay Sinir Ağı	7
2.2.2.1. Çok Algılayıcı Sinir Ağı	8
2.2.2.2. Derin Öğrenme	9
2.2.2.3. Yinelenen Sinir Ağı	9
2.3. BULUT BİLİŞİM	10
2.3.1. Servis Modelleri	11
2.3.2. Konumlandırma Modelleri	13
3. MALZEME VE YÖNTEM	15
3.1. VERİ SETLERİ	15
3.1.1. Mevcut Veri Seti	16
3.1.2. Oluşturulan Veri Seti	18
3.2. MAKİNE ÖĞRENİMİ	19
3.2.1. Modelin Yapısı	19

3.2.2.	Modelin Oluşturulması	22
3.2.2.1.	<i>MATLAB ile Model Oluşturma</i>	22
3.2.2.2.	<i>TensorFlow ile Model Oluşturma</i>	25
3.2.2.3.	<i>Keras ile Model Oluşturma</i>	25
3.3.	MOBİL YAZILIM	25
3.3.1.	Veriye Erişim	25
3.3.1.1.	<i>Dahili Algılayıcılar</i>	27
3.3.1.2.	<i>Civardaki Cihazlarla Haberleşme</i>	28
3.3.1.3.	<i>Veritabanı Üzerinden Senkronizasyon</i>	29
3.3.2.	Mobil Ortamda Makine Öğrenimi	30
3.3.2.1.	<i>Önceden Eğitilmiş Model</i>	31
3.3.2.2.	<i>Kullanıcıya Adapte Olan Model</i>	32
3.3.3.	Mobil Arayüz Tasarımı	33
3.4.	BULUT YAZILIMI	33
3.4.1.	Kullanıcı ve Veri Yönetimi	35
3.4.2.	İnternet Sayfası Arayüzü Tasarımı	36
3.4.3.	Sunucu Üzerinde Makine Öğrenimi	36
3.4.3.1.	<i>Modelin Eğitilmesi</i>	37
3.4.3.2.	<i>Modelin Hizmete Sunulması</i>	38
4.	BULGULAR	40
4.1.	MAKİNE ÖĞRENİMİNİN BAŞARISI	40
4.1.1.	Performans Ölçüm Yöntemi	40
4.1.2.	Mevcut Veri Setinin Sonuçları	41
4.1.3.	Oluşturulan Veri Setinin Sonuçları	42
4.2.	SİSTEM MİMARİSİNİN DEĞERLENDİRİLMESİ	44
5.	TARTIŞMA VE SONUÇ	47
5.1.	GENEL SONUÇ	47
5.2.	GELECEK ÇALIŞMALAR	48
KAYNAKLAR	49
EKLER	51
EK 1.	Oluşturulan veri seti için yapılan tahminler	51
ÖZGEÇMİŞ	56

ŞEKİL LİSTESİ

	Sayfa No
Şekil 2.1: Doğrusal regresyon modeli.	6
Şekil 2.2: Tek algılayıcıdan oluşan yapay sinir ağı modeli.	7
Şekil 2.3: Bir katmanda birden fazla algılayıcı barındıran yapay sinir ağı modeli.	8
Şekil 2.4: Derin yapay sinir ağı modeli.	9
Şekil 2.5: Yinelenen sinir ağının açılımı.	10
Şekil 3.1: Veri kümelerinin hazırlanmasında kullanılan sözde kod.	20
Şekil 3.2: Makine öğreniminde kullandığımız akış diyagramı.	21
Şekil 3.3: Yinelenen sinir ağı modeli.	22
Şekil 3.4: Modelin eğitilmesi ve test edilmesinde kullanılan sözde kod.	23
Şekil 3.5: Modelin eğitiminde yapılan işlemlerin akış grafiği.	24
Şekil 3.6: Adım sayısının tahmininde yapılan işlemlerin akış grafiği.	24
Şekil 3.7: Mobil cihazla veriye erişim.	26
Şekil 3.8: Veri erişimi yetki ekranı.	29
Şekil 3.9: Önceden eğitilmiş model kullanımında sunucu ile haberleşme.	31
Şekil 3.10: Kullanıcıya adapte olan model kullanımında sunucu ile haberleşme.	32
Şekil 3.11: Mobil yazılım ekran görüntüsü.	34
Şekil 3.12: İnternet uygulamasına giriş ekranı.	36
Şekil 3.13: İnternet uygulaması ekran görüntüsü.	37
Şekil 3.14: Modelin sunucu üzerinde hizmet sunması.	38
Şekil 4.1: En düşük hatalı veri kümesi.	43
Şekil 4.2: Referans yönteme kıyasla en başarılı veri kümesi.	43
Şekil 4.3: Test verisindeki varyansın performans sonuçlarına etkisi.	44

TABLO LİSTESİ

	Sayfa No
Tablo 3.1: StudentLife veri setinden fiziksel aktivite verisi örneği.	17
Tablo 3.2: Xiaomi Mi Band 2 cihazına ait teknik özellikler.	19
Tablo 4.1: Mevcut veri seti ile elde edilen sonuçlar.	41
Tablo 4.2: Oluşturulan veri seti ile elde edilen sonuçlar.	42
Tablo 4.3: Alternatif çözümlerin özellik tablosu.	45

SİMGE VE KISALTMA LİSTESİ

Kısaltmalar	Açıklama
API	: Application Program Interface (Uygulama Programı Arayüzü)
BLE	: Bluetooth Low Energy (Düşük Enerjili Bluetooth)
GPS	: Global Positioning System (Küresel Konumlama Sistemi)
NIST	: National Institute of Standards and Technology (Amerika Birleşik Devletleri Ulusal Standartlar ve Teknoloji Enstitüsü)
UTC	: Coordinated Universal Time (Eş Güdümlü Evrensel Zaman)

ÖZET

YÜKSEK LİSANS TEZİ

SAĞLIK VERİLERİNİN ANALİZİ İÇİN MOBİL VE BULUT YAZILIMI TASARIMI

Gökhan ÖZOĞUR

İstanbul Üniversitesi

Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Dr. Öğr. Üyesi Muhammed Ali AYDIN

Günümüzde her an yanımızda taşıdığımız akıllı telefonlar sayesinde sağlık durumumuz hakkında bilgi verebilecek birçok verinin toplanabilmesi mümkündür. Akıllı telefonlar hem kendi içerilerinde barındırdıkları algılayıcılardan hem de etraflarındaki çeşitli cihazlar ile haberleşerek onların içerisinde bulunan algılayıcılardan elde ettikleri verileri kullanarak sağlık durumumuz hakkında bilgi verebilmektedir. Sağlık verilerinin takip amaçlı izlenmesinin yanı sıra işlenerek anlamlı bilgiler sağlanması mümkündür.

Bu tez çalışmasında sağlık verileri kullanılarak kişilerin gelecekteki sağlık durumlarını tahmin edebilecek mobil ve bulut tabanlı bir sistemin tasarlanması amaçlanmıştır. Bu amaç doğrultusunda sağlık verilerini toplayacak bir mobil uygulama geliştirilmiştir. Akıllı bileklik ve telefonda gelen veriler analiz edilerek kullanıcıların gelecek zaman dilimlerinde gerçekleştirecekleri fiziksel aktivite seviyeleri tahmin edilmektedir. Makine öğrenimi yöntemi kullanılarak oluşturulan modelin mobil ve bulut yazılımlar üzerinde çalıştığı alternatif çözümler sunulması kullanım senaryoları karşılaştırılmıştır. Oluşturulan modelin performansının referans yönteme göre daha başarılı olduğu tespit edilmiştir.

Haziran 2018, 67 sayfa.

Anahtar kelimeler: veri analizi, makine öğrenimi, bulut bilişim, mobil sağlık.

SUMMARY

M.Sc. THESIS

DESIGN OF MOBILE AND CLOUD SOFTWARE FOR ANALYSIS OF HEALTH DATA

Gökhan ÖZOĞUR

İstanbul University

Institute of Graduate Studies in Science and Engineering

Department of Computer Engineering

Supervisor: Dr. Öğr. Üyesi Muhammed Ali AYDIN

Nowadays it is possible to collect a lot of data that can give us information about our health condition using the smartphones we carry with us at any moment. Smartphones provide information about our health situation using the data both they get from the sensors in them and the sensors in various devices around them. In addition to tracking the health data, processing them provides significant information about our health situation.

In this thesis study, it is aimed to design a mobile and cloud based system that can predict the future health status of people by using the health data. For that purpose, a mobile application has been developed to collect health data. Data from the smart bands and phones are analyzed to predict the future physical activity levels of the users. Alternative solutions that machine learning model works on mobile and cloud software are presented and the use-case scenarios are compared. As a result, it has been found that the performance of the designed model is much more successful than the reference method.

June 2018, 67 pages.

Keywords: data analytics, machine learning, cloud computing, mobile health.

1. GİRİŞ

Günümüzde akıllı telefonların geniş kitleler tarafından kullanılmaya başlaması ile mobil sağlık uygulamalarına olan talep artmaktadır. Her an yanımızda taşıdığımız akıllı telefonlar sayesinde sağlık durumumuz hakkında bilgi verebilecek birçok verinin toplanabilmesi kolaylaşmıştır. Akıllı telefonlar hem kendi içerilerinde barındırdıkları algılayıcılardan hem de etraflarındaki çeşitli cihazlar ile haberleşme yoluyla onların içerisinde bulunan algılayıcılardan elde ettikleri verileri kullanarak sağlık durumumuz hakkında bilgi verebilmektedir. Elde edilen kişisel sağlık verilerinden yararlanarak hasta bazlı sağlık çözümlerinin uygulanması mümkündür.

Bu tez ile amaçlanan, kişilerden toplanacak sağlık verilerinin analiz edilmesi sonucu kişiye özel tavsiye niteliğinde geri bildirimlerde bulunabilecek bir sistemin tasarlanmasıdır. Prestwich ve çalışma arkadaşları tarafından yapılan araştırmada kullanıcılara fiziksel aktivitede bulunmaları için içinde buldukları duruma özel hatırlatmalarda bulunmanın fiziksel aktivite seviyesini arttırdığı sunulmaktadır [1]. Bu bakımdan kişiye doğru zamanda geri bildirimde bulunulması daha sağlıklı bir yaşam elde edilmesi için önem taşımaktadır. Kişisel sağlık verilerinin analizi, takibi ve gelecekteki sağlık durumunun tahmini bir sistem tasarımına ihtiyaç vardır. Bu tezde böyle bir sistemin hayata geçirilmesinde kullanılacak mobil ve bulut yazılım tabanlı bir tasarımının ortaya konulması amaçlanmıştır.

Tez kapsamında geliştirdiğimiz mobil uygulama yardımıyla kullanıcıların sağlık verileri toplanmaktadır. Bu sağlık verilerinin analiz edilmesi sonucunda ortaya çıkan makine öğrenimi modeli ile gelecek zamanlardaki fiziksel aktivitede bulunma oranları tahmin edilmektedir. Akıllı bileklik ve telefonda gelen veriler ile kullanıcılara gün içerisinde attıkları adım sayısı ve bir sonraki saat gerçekleştirmesi beklenen tahmini değer sunulmaktadır. Tasarlanan sistem ile hem mobil hem de internet sayfası arayüzü üzerinden sağlık durumunun takip edilmesi ve geri bildirim alınması mümkündür. Kişilerin kendi sağlık durumlarının takibini yapabilmelerinin yanı sıra yetki verilen akraba ve sağlık personeli gibi kişilerin de sağlık durumunun takibini yapabilmeleri mümkündür.

Sağlık durumunun farklı cihazlardan ve yetki verilen diğer kişilerce takibinin yapılabilmesi için bir sunucuya ihtiyaç duyulmaktadır. Bu sunucu üzerinde çalışacak olan yazılım ile

sağlık verilerinin depolanması, yetkili kişilere erişim sağlanması, kullanıcıların yönetilmesi ve verilerin işlenerek gelecek değerlerin tahmin edilmesi işlemleri gerçekleştirilmelidir. Bunların yanı sıra sunucunun bakımı ve güvenliğinin sağlanması gibi bir takım yönetsel işlerin de olduğu göz önüne alındığında sistem tasarımında bulut bilişimin tercih edilmesi ön plana çıkmaktadır. İhtiyaç duyulan bilişim kaynağının istenilen kapasitede kiralanabilmesi ve ücretlendirmenin kullanıma göre yapılması bulut bilişimin en önemli avantajlarından. Ayrıca çok sayıdaki kullanıcının aynı anda sistemi kullanmak istemesi durumunda sistemde oluşacak yüklenmeleri azaltmak amacıyla bilişim kapasitesinin anlık olarak ölçeklendirilebilmesi bulut bilişim ile mümkün olmaktadır. Tez kapsamında geliştirdiğimiz yazılım ile sağlık verilerinin bulut üzerinde yedeklenmesi, senkronizasyonu ve yetki verilen kişilere servis edilmesi sağlanmaktadır. Ayrıca makine öğrenim algoritması bulut üzerinde çalıştırılarak modelin eğitilmesi ve hizmet sunması gerçekleştirilmiştir.

İhtiyaç duyulan tasarımın gerçekleştirilmesi için mobil ve bulut yazılım tasarımları gerçekleştirilmiştir. Bu tasarımlar içerisinde mobil ağırlıklı ve bulut ağırlıklı farklı çözümler yer almaktadır. Gereksinimlere uygun bir çözümün uygulanabilmesi için yapılan işlemler tez boyunca aktarılmıştır. Bu kapsamda oluşturulan tezin yapısı aşağıda sunulmaktadır.

Genel Kısımlar bölümünde mobil sağlık alanında yapılan çalışmalar sunularak mobil sağlığın önemi vurgulanmaktadır. Ayrıca dijital ortamda saklanan sağlık verileri sunulmakta ve veri analizinde kullanılan yöntemlerden bahsedilmektedir. Makine öğrenimi modelini oluştururken kullandığımız yapay sinir ağları hakkında bilgiler bu bölümde verilmektedir. Bölümün sonunda ise önerilen sistem tasarımının önemli bir parçası olan bulut bilişim hakkında bilgi verilmektedir ve bulut bilişim çözümlerindeki seçim kriterleri olan servis tipi ve konumlandırma modelleri açıklanmaktadır.

Malzeme ve Yöntem bölümünde veri analizinde kullandığımız veri setleri ve makine öğrenimi yöntemleri açıklanmaktadır. Verilerin hazırlanması ve analiz edilmesi yapılan işlemler ve makine öğreniminde kullandığımız algoritma hakkında bilgi bu bölümde aktarılmaktadır. Ayrıca bu makine öğreniminin mobil ve bulut platformlarında nasıl uygulanacağı anlatılmaktadır. Bölümün sonunda ise kullanıcı verilerine erişim ve analiz aşamalarında uygulanabilecek alternatifler sunularak mobil ve bulut yazılım tasarımı ortaya çıkarılmaktadır.

Bulgular bölümünde tez kapsamında oluşturulan makine öğrenimi modeli ile elde edilen sonuçlar gösterilmektedir. Oluşturduğumuz model ile referans model karşılaştırılarak performans sonuçları değerlendirilmektedir. Ayrıca sistem mimarisinde uygulanabilecek alternatif yöntemlerin avantaj ve dezavantajları aktarılmaktadır.

Tartışma ve Sonuç bölümünde tez kapsamında elde edilen sonuçlar değerlendirilmekte ve gelecekte yapılabilecek çalışmalar sunulmaktadır.



2. GENEL KISIMLAR

2.1. MOBİL SAĞLIK

2.1.1. Mobil Sağlık Alanında Yapılmış Çalışmalar

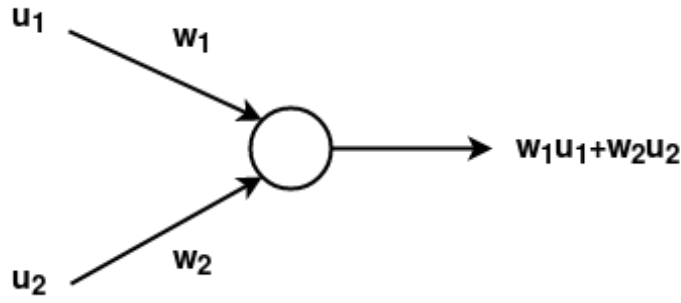
Bireylerin sağlık durumları hakkında fikir sahibi olabilmek için yaptıkları fiziksel aktivitelerle ilgili bilgiye sahip olmak gerekmektedir. Bu bilgiye mobil cihazlardan toplanan fiziksel aktivite verilerini işleyerek ulaşabiliriz. Örnek vermek gerekirse sağlıklı yaşama ait önemli göstergelerden biri olan uyku kalitesi hakkında tahmin yapabilmek için fiziksel aktivite verileri kullanılabilir. Bireylerin almaları gereken günlük uyku sürelerini gün içerisinde yaptıkları fiziksel aktivite seviyesine göre belirleyen çalışmalar mevcuttur. Sathyanarayana ve çalışma arkadaşları tarafından yapılan bir çalışmada uyku sürelerinin hesaplanması için bireylerden gün boyunca toplanan fiziksel aktivite verileri doğrusal regresyon ve derin öğrenme modelleri ile birlikte kullanılmaktadır [2]. Sarwar ve çalışma arkadaşları tarafından yapılan farklı bir çalışmada ise adım sayısı verisi kullanılarak bireylerin gerçekleştirdikleri günlük fiziksel aktivite süreleri hesaplanmaktadır [3]. Bireyler fiziksel aktivite performanslarına bağlı olarak aynı çalışma kapsamında geliştirilen SmartFit isimli uygulama üzerinde puan kazanmaktadır. Oyunlaştırma adı verilen bu teknikte oyun dışı problemler bir oyun tasarımının parçası gibi ele alınarak çözülmektedir [4]. SmartFit uygulamasında kullanılan oyunlaştırma tekniği sayesinde kullanıcılar spor yapmak için motive edilmektedir. Consolvo ve çalışma arkadaşları tarafından yapılan benzer bir uygulamada kullanıcılar için fiziksel aktivite hedefleri belirlendikten sonra ilerlemelerine bağlı olarak puan kazanmaları sağlanmaktadır [5]. Fakat sadece bu tekniğin uygulanması bireylerin uzun süre hareketsiz kalmasını engellemek için yeterli olmamaktadır. Uzun süreli hareketsizliği engelleyebilmek için bir takip ve uyarı sistemine ihtiyaç vardır.

Yeni bir alışkanlığın kazandırılabilmesi için mevcut alışkanlıkları inceleyip uygun olan durumlarda uyarıda bulunacak bir sistem tasarımının yapılması gerekmektedir. Bu kapsamda yapılmış çalışmalar arasında Fitbit cihazını kullanan kullanıcıları gerçek zamanlı olarak takip eden bir uyarı sistemi tasarımı bulunmaktadır [6]. Pina ve çalışma arkadaşları tarafından önerilen sistemde, kullanıcılar belirlenmiş zaman aralıklarında gerekli miktarda fiziksel

aktivitede bulunmadıklarında bir bildirim almaktadır. Eğer kullanıcı kısa bir zaman içinde herhangi bir aksiyonda bulunmazsa ikinci uyarı gönderilmektedir. Bu sistem hareketsiz kalınan sürelerin farkında olunmasını sağlamaktadır fakat fiziksel aktivitede bulunulması istenen zamanlarda kullanıcıların ne kadar müsait olduğu değerlendirilmediği için uyarıların uygulanabilirliği olmayabilir. Kullanıcı alışkanlıklarını inceleyip onlara uygun zamanda uyarıda bulunacak bir sistem bireyler tarafından kolayca kabul görecektir. Bireylerin alışkanlıklarının öğrenilmesi sonucunda gelecekte benzer alışkanlıkları ne zaman tekrar edecekleri tahmin edilebilecektir. Geçmiş fiziksel aktivite verilerini kullanarak bireylerin gelecek zamanlardaki fiziksel aktivitede bulunma oranını tahmin eden çalışmalar bulunmaktadır. He ve Agu tarafından yapılan bir çalışmada belirli zaman aralıklarında kullanıcıların hareketsiz kalma oranları hesaplanmakta ve oto-regresyon modeli kullanılarak gelecek zaman periyodu için hareketsiz olma oranı hesaplanmaktadır [7]. Benzer bir çalışma farklı bir yöntem kullanılarak tez sürecinde bizim tarafımızdan da yapılmıştır [8]. Yaptığımız çalışmada aynı veri seti derin öğrenme yöntemi ile birlikte kullanılmıştır. Sunmuş olduğumuz derin öğrenme modeliyle gelecek zaman periyotları için hareketsiz olma oranları tahmini yapılmaktadır. Bu çalışmalarda uygulanan yöntemlerle bireylerin hangi zaman aralıklarında aktivite yapmak için daha müsait olacağı tahmin edilebilmektedir.

2.1.2. Sağlık Verileri

Apple şirketi tarafından üretilen mobil cihazlarda bulunan HealthKit adı verilen yazılım geliştirme paketi sayesinde kullanıcılara ait boy, ağırlık, vücut kitle endeksi, vücut yağ oranı, kalp atış hızı, vücut sıcaklığı, kan basıncı, kan şekeri seviyesi, solunum hızı, aldığı insulin miktarı, egzersiz sırasındaki maksimum oksijen tüketimi, uyku analizi ve beslenme alışkanlığı gibi bilgilere erişilebilmektedir [9]. Bunların yanı sıra kullanıcıların fiziksel aktiviteleri hakkında daha detaylı bilgi edinilmesi için adım sayısı, yürüme ve koşma mesafesi, bisiklet ile alınan mesafe, tekerlekli sandalye ile alınan mesafe ve itiş sayısı, yüzme mesafesi ve kulaç sayısı, tırmanılan basamak sayısı, yakılan enerji miktarı, hareket edilen süre ve ayakta geçirilen süre gibi bilgilerin takibi yapılmaktadır.



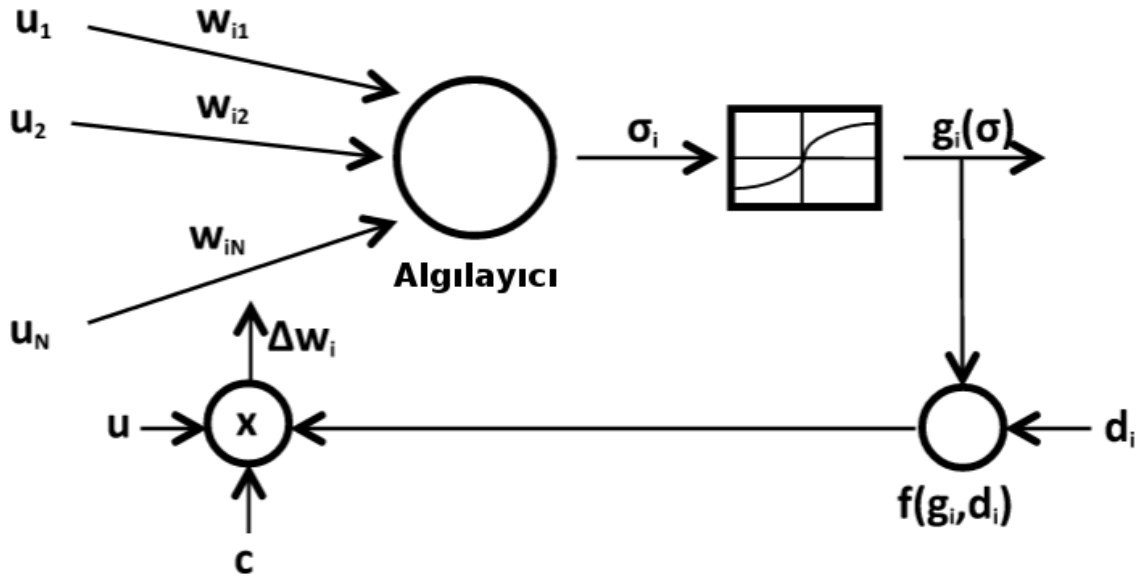
Şekil 2.1: Doğrusal regresyon modeli.

2.2. VERİ ANALİZİ

Veri analizi, fayda sağlayacak bir bilgiye ulaşmak amacıyla eldeki verilerin incelenmesi işlemidir. Bu amaçla veri toplama, işleme ve temizleme işlemleri yapıldıktan sonra elde kalan veriler kullanılarak bir veri modeli oluşturulmaktadır. Bu modele sahip olunan veriler girdi olarak verilerek verilerin anlamlandırılması sağlanmaktadır. Ayrıca yeni elde edilecek verilerin kullanılması ile gelecek sonuçların tahmini yapılabilmektedir. Model bir fonksiyon olarak ele alınırsa, yeterli sayıda girdinin fonksiyona verilmesi sonucu elde edilecek çıktı ile istenilen sonuç arasındaki hata karşılaştırılarak fonksiyonun karakteristiği elde edilebilmektedir. Benzer şekilde fonksiyonun karakteristiğini etkileyen parametreler değiştirilerek modelden istenilen sonuçlar çıktı olarak alınabilmektedir. Verinin ve dolayısıyla modelin karmaşıklığına bağlı olarak iyi sonuçlar elde edebilmek için kullanılacak parametrelerin sayısı değişmektedir. Modeli daha iyi ifade edebilmek amacıyla regresyon ve yapay sinir ağları gibi yapılar kullanılmaktadır.

2.2.1. Doğrusal Regresyon

Basit bir veri modelinin doğrusal regresyon ile ifade edilmesi mümkündür. Doğrusal regresyon, bir ya da daha fazla sayıdaki bağımsız değişken ile bu değişkenlere bağımlı bir değişken arasındaki ilişkinin doğrusal olarak ifade edilmesidir. Veri modeli üzerinden açıklamak gerekirse modele girdi olarak verilen değerlerin doğrusal kombinasyonları ile bir çıktının elde edilmesi işlemidir. Şekil 2.1'de girdiler u ve girdilere ait katsayılar w simgesi ile gösterilmektedir.

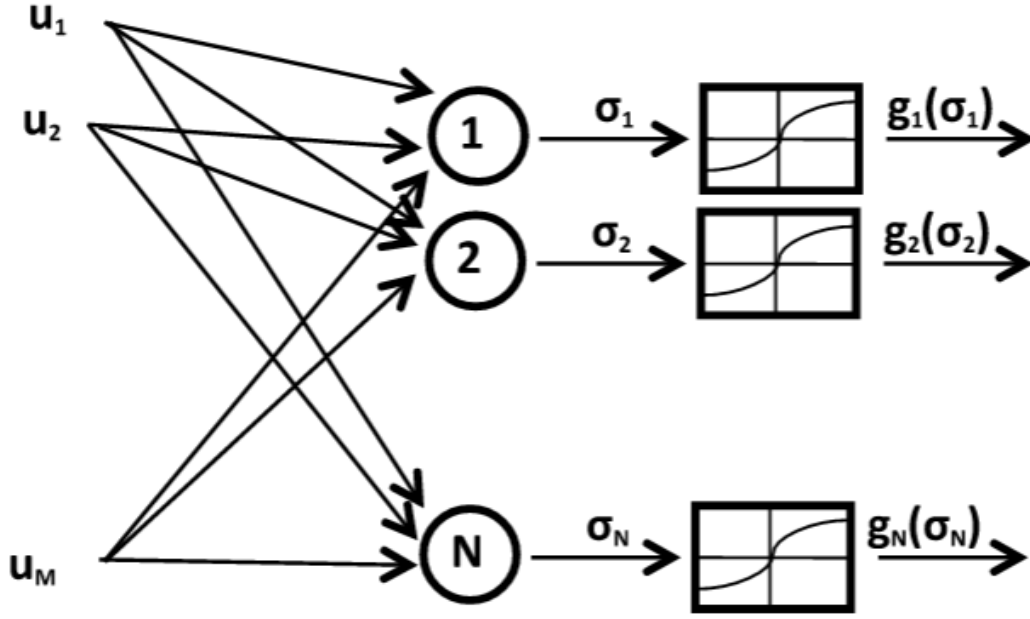


Şekil 2.2: Tek algılayıcıdan oluşan yapay sinir ağı modeli.

2.2.2. Yapay Sinir Ağı

Yapay sinir ağlarında hesaplama işlemi algılayıcılar (nöronlar) üzerinden gerçekleşmektedir. Algılayıcı kendisine ulaşan her bir girdi için bir adet ağırlık katsayısı denilen çarpan barındırmaktadır. Bir algılayıcıya gelen girdilerin tümü ilgili ağırlık katsayısı ile çarpıldıktan sonra toplanmaktadır. Elde edilen bu toplam sonuca algılayıcının net etkisi denmektedir. Net etki değerinin aktivasyon fonksiyonundan geçirilmesi sonucu algılayıcının çıktı değeri elde edilmektedir. Belirlenen aktivasyon fonksiyonuna göre algılayıcının ne derece aktif çalışacağı belirlenmektedir.

Her sistemde olduğu gibi yapay sinir ağı sisteminde de amaç elde edilen çıktı ile istenilen sonuç arasındaki farkın en aza indirilmesidir. Yapay sinir ağlarında hatanın en aza indirilmesi için gerekli olan eğitim işlemi ağırlık katsayılarının güncellenmesi ile gerçekleşmektedir. Eğitici öğrenme yöntemlerinde algılayıcıdan elde edilen çıktı ile istenilen sonuç arasındaki fark hesaplanmaktadır ve buna göre her bir katsayı için güncelleme yapılmaktadır. Şekil 2.2'de yapay sinir ağında öğrenme süreci gösterilmektedir. Şekil 2.2'de N adet girdi u , N adet ağırlık katsayısı w , net etki değeri σ , çıktı değeri g , istenilen sonuç d , öğrenme katsayısı ise c simgesi ile gösterilmektedir.

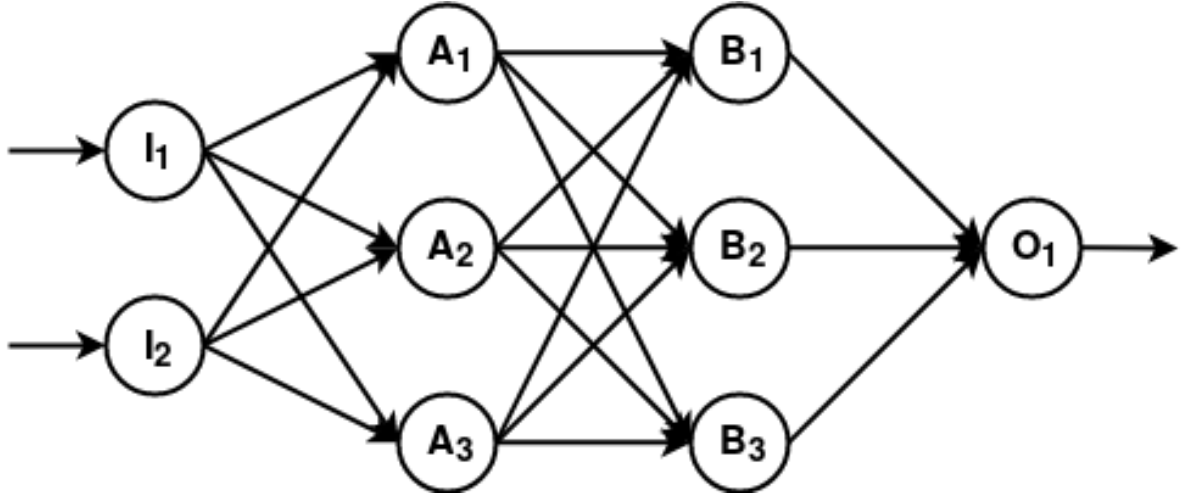


Şekil 2.3: Bir katmanda birden fazla algılayıcı barındıran yapay sinir ağı modeli.

2.2.2.1. Çok Algılayıcı Sinir Ağı

Yapay sinir ağlarında tipik olarak birden fazla algılayıcı kullanılarak bir ağ modeli oluşturulmaktadır. Çok algılayıcıdan oluşan bu modellerde girdiler birden fazla algılayıcı tarafından kullanılmaktadır. Her algılayıcı kendine ait ağırlık katsayılarını barındırdığından dolayı bir girdi farklı algılayıcılar için farklı etkiler gösterebilmektedir. Şekil 2.3'te çok sayıda algılayıcıdan oluşan bir yapay sinir ağının işleyişi gösterilmektedir.

Çok sayıda algılayıcı barındıran bir katmanlı yapay sinir ağında algılayıcı sayısı kadar çıktı elde edilmektedir. İstenilen çıktı sayısına göre algılayıcı sayı ayarlanabilir fakat bu durumda sistem performansı oldukça kısıtlanmaktadır. Örneğin istenilen çıktı sayısı bir ise tek algılayıcı kullanmak istenilen sonucu vermeyebilir. Bu duruma çözüm olarak birden fazla algılayıcı katmanı oluşturulmaktadır. Girdi katmanı olarak adlandırılan ilk katman girdilere erişimi olan algılayıcılardan oluşmaktadır. İstenilen çıktı sayısı kadar algılayıcıdan oluşan son katman ise çıktı katmanı olarak adlandırılmaktadır. Girdi katmanındaki algılayıcıların çıktıları kendinden sonra gelen katmandaki algılayıcılar için girdi olarak ele alınmaktadır. Böylece tek katmanlı yapıdaki işleyiş çok katmanlı yapıda da sürdürülmektedir.



Şekil 2.4: Derin yapay sinir ağı modeli.

2.2.2.2. Derin Öğrenme

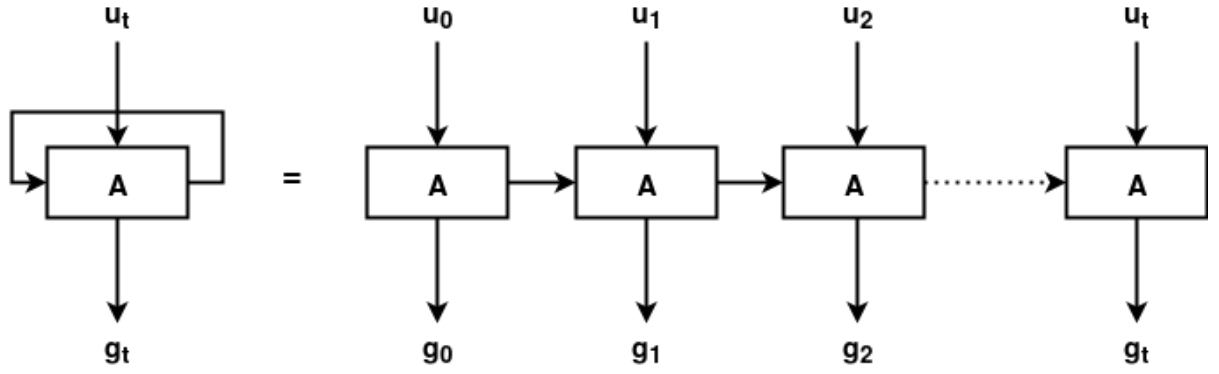
Çok katmanlı yapay sinir ağlarında girdi katmanı ile çıktı katmanı arasına gizli katman adı verilen bir algılayıcı katmanı eklenmesi mümkündür. Gizli katmanın eklenmesi ile yapay sinir ağı modeli daha derin bir yapıya sahip olmaktadır. Birden fazla gizli katman içeren yapay sinir ağı modellerine derin yapay sinir ağları denmektedir. Bu modellerde gerçekleştirilen öğrenme işlemleri ise derin öğrenme olarak adlandırılmaktadır.

Şekil 2.4'te 2 adet girdi alıp 1 adet çıktı veren 2 gizli katmanlı bir derin yapay sinir ağı örneği gösterilmektedir. Şekil 2.4'te girdi katmanındaki algılayıcılar I , gizli katmanlardaki algılayıcılar A ve B , çıktı katmanındaki algılayıcı O harfi ile ifade edilmektedir.

2.2.2.3. Yinelenen Sinir Ağı

Geleneksel yapay sinir ağlarında öğrenme süreci bittikten sonra sisteme defalarca aynı girdi verilirse sistemden alınacak çıktı aynı olmaktadır. Bir girdinin farklı bir girdiden önce ya da sonra verilmesi arasında elde edilen sonuç bakımından bir fark olmamaktadır. Bu yapı zaman serisi gibi birbirinden tamamen bağımsız olmayan girdiler kullanıldığında yetersiz kalmaktadır. Örnek vermek gerekirse bir cümlenin içerisinde yer alan kelimeler arasında bile bir ilişki bulunmaktadır. Kelimelerin sıralaması bilindiğinde bir sonraki kelimeyi tahmin edebilmek daha kolay olmaktadır.

Yinelenen sinir ağı modelleri kullanılarak önceki girdilerden elde edilen bilgi birikiminin sonraki girdilere aktarılması mümkün olmaktadır. Bu nedenle modele verilen girdilerin



Şekil 2.5: Yinelenen sinir ağının açılımı.

sıralaması önem kazanmakta ve modelin davranışını etkilemektedir. Bilgi birikiminin aktarılması katmanların hafızaları üzerinden yapılmaktadır. Şekil 2.5'te yinelenen sinir ağına ait bir katmandaki döngü gösterilmektedir. Şekil 2.5'te sinir ağı katmanı A, girdi u , sonuç ise g harfi ile ifade edilmektedir. Sinir ağı katmanına verilen girdi hem sonucu belirlemede hem de katmanın durumunu değiştirerek bir sonraki girdi verildiğinde farklı davranmasını sağlamaktadır.

2.3. BULUT BİLİŞİM

Bulut bilişim kullanıcıların veriye daha az maliyetle ve daha hızlı bir şekilde ulaşmasını sağlamak amacıyla veri ve uygulamaları muhafaza etmek, işlemek ve kullanmak için internet ağını ve uzak sunucuları kullanan servis tabanlı bir teknolojidir. NIST tarafından yapılan tanıma göre bulut bilişim; bilgisayar ağları, sunucular, veri depolama alanları, uygulamalar ve hizmetler gibi yapılandırılabilir bilişim kaynaklarından oluşan ortak bir havuza uygun koşullarda ve isteğe bağlı olarak herhangi bir zamanda herhangi bir yerden erişime olanak veren bir modeldir [10]. Bu modelde sistem kaynakları yönetimsel çaba bakımından diğer bilişim kaynaklarına göre daha düşük seviyede etkileşim gerektirmektedir. NIST tarafından verilen bulut bilişim tanımına göre bir bulut bilişim platformunun temel olarak sahip olması gereken bazı karakteristik özellikler vardır. Bu özellikler ihtiyaca göre belirlenebilir kaynaklar, geniş ağ erişimi, kaynak paylaşımı, anlık esneklik ve ölçülebilir servis olarak belirtilmektedir [10].

Bulut bilişim platformlarında sunucu işlem zamanı ve depolama alanı gibi bilişim kapasiteleri ihtiyaç duyulduğu anda kullanıcı tarafından otomatik bir şekilde artırılıp azaltılabilmelidir. Bu sayede kullanıcılar uygulamalar için ihtiyaç duyulan altyapının ne kadar güçlü olacağını ve ne

kadar kaynak kullanacağını önceden kesin bir şekilde analiz etmeden yönetme imkanına sahip olmaktadır.

Bulut bilişim kaynakları kullanıcılara ağ üzerinden dizüstü bilgisayarlar, tabletler ve mobil telefonlar gibi farklı platformlardan erişilebilecek şekilde sunulmaktadır. Bu nedenle bir bulut bilişim platformu farklı tipteki cihazlara hizmet edebilecek geniş bir ağ erişimine sahip olmalıdır.

Fiziksel ve sanal kaynaklar bulut bilişim hizmet sağlayıcısı tarafından birden fazla kullanıcının kiralayabileceği bir kaynak havuzunda sunulmaktadır. Çoklu kiralama mimarisinde çok sayıdaki kullanıcının aynı altyapıyı kullanılabilmesine olanak sağlanmaktadır. Kullanıcı kendisine sağlanan kaynakların tam olarak nerede olduğu ile ilgili bilgiye sahip olmadan sadece ülke, bölge veya veri merkezi bazında seçim yaparak ihtiyaç duyduğu kaynağı kullanabilmektedir.

Bulut bilişim kullanıcısının talebi veya sistem kullanım durumuyla orantılı olarak ihtiyaç duyulan kaynaklar için bilişim kapasitesi anında esnek bir şekilde artırılabilir. Benzer bir durumda kullanım ihtiyacı sona erdiğinde kullanılmayan kaynakların kapasitesi azaltılabilir. Bu sayede kullanıcı ihtiyacı olduğu anda ihtiyacı kadar kapasiteyi kiralarak sadece kullandığı kaynak için ödeme yapmaktadır.

Bulut bilişimde kaynak kullanım miktarları otomatik olarak ölçülerek kontrol altında tutulmaktadır. Depolama alanı, işlemci gücü, bant genişliği ve aktif kullanıcı hesabı sayısı gibi parametreler hem kullanıcılar hem de bulut bilişim sağlayıcısı için şeffaf bir şekilde raporlanmaktadır. Böylece kullanıcılar kullandıkları kaynakları daha aza indirilebilmek için kontrol sahibi olmaktadır.

2.3.1. Servis Modelleri

Teknik yeterlilik ve ilgi bakımından farklı seviyedeki kullanıcıların ihtiyaçlarına yönelik farklı servis modelleri ile bulut bilişim hizmeti sunulmaktadır. Yazılım servisi, platform servisi ve altyapı servisi olarak 3 temel servis modeli bulunmaktadır. Bu modellerin seçiminde kullanıcılar ihtiyaçlarına uygun olarak bulut servislerinin yönetimi için gerekli çaba ve zaman ile platform üzerindeki yetki düzeyi arasında bir seçim yapmaktadır. Sistem üzerinde sadece belirli alanlarda yetkiye ihtiyaç duyan bir kullanıcı en az düzeyde çaba sarf ederek bulut bilişim hizmetinden

yararlanabilmektedir. Buna zıt olarak sistem üzerinde tüm kontrolü elinde bulundurmak isteyen bir kullanıcı daha alt seviyeden bir bulut servisini kullanmayı tercih etmektedir.

Yazılım servis modeli, sunucu üzerinde bulundurulan yazılım uygulamalarının bulut bilişim sağlayıcısı tarafından birden fazla kişi veya kuruluşun kullanımına sunulmasıdır. Bu servis modelinde kullanıcılar en üst seviye olan yazılım katmanında kiralama yapmaktadırlar. Bu nedenle çoğu kaynak kullanıcılar tarafından ortaklaşa kullanılmakta ve lisans bedelleri paylaşılmaktadır. Kullanıcılar yazılımın kurulumu, lisanslaması ve bakımı gibi sorunlarla uğraşmamakta ve böylelikle bu işler için gerekli olan maliyet ve kaybedilen zaman da kendiliğinden ortadan kalkmaktadır. Kullanıcılar bu modelde düşük maliyetle bilişim hizmetlerini kullanabilmektedir fakat sistem üzerindeki yetkileri sadece yazılım katmanı ile sınırlıdır. Bunun yanında uygulama yönetimi dışında sistemin geri kalan kısımlarının güncellenmesi, bakımı ve güvenliği gibi konularda bilgi sahibi olma gereksinimi ortadan kalkmaktadır. Kullanıcılar uygulamalara internete bağlı herhangi bir bilgisayar veya mobil cihazla erişebilmekte ve yönetebilmektedir.

Platform servis modeli, bulut bilişim sağlayıcısı tarafından kullanıcıların kendi geliştirdikleri uygulamaları çalıştırabileceği bir platformun ve gerekli olan tamamlayıcı altyapılarla birlikte birden fazla kişi veya kuruluşun kullanımına sunulmasıdır. Kullanıcıların kendi kurduğu uygulamalar dışında platform altyapısını oluşturan bileşenler üzerinde herhangi bir yönetim ve kontrol olanağı bulunmamaktadır. Kullanıcılar uygulamalarının geliştirilmesi ve çalışması sırasında gerekli olan işletim sistemi, uygulama ve donanımların temin edilmesi için uğraşmadan bulut bilişim sağlayıcısı tarafında sağlanan hizmetten yararlanmaktadır. Platform servis modeli yazılım servis modeline göre daha fazla yetki vermektedir. Kullanıcılar platform katmanında yetki sahibi olmasına rağmen sunulan altyapı ve donanım üzerinde değişiklik yapma yetkisi bulunmamaktadır. Buna rağmen donanım satın alma maliyeti olmadığı için avantajlı bir kullanım imkanı sunmaktadır.

Altyapı servis modeli, kullanıcıların ihtiyaç duydukları donanım, ağ ekipmanı ve depolama birimi gibi temel bilişim kaynaklarını yapılandırabildiği ve bu altyapı üzerinde istedikleri işletim sistemi, araç ve uygulamaları kurabildikleri kiralama modelidir. Kullanıcıların alt yapıya doğrudan erişimi bulunmamaktadır fakat sistem üzerinde işletim sistemi seviyesinde yetkisi bulunmaktadır. Bulut bilişim sağlayıcıları sanallaştırma teknolojisi sayesinde kendi bünyelerindeki fiziksel sunucularda sanal makinalar oluşturarak bu makinaları kiralamaktadır.

Bu teknikle birbirinden izole edilmiş sanal sunucu makineleri oluşturularak fiziksel olarak sahip olunandan çok daha fazla sayıdaki sanal sunucu kiralanabilmektedir. Bu servis modelinde ihtiyaç duyulan kaynaklar kullanıcılar tarafından fiziksel olarak satın alınmadığı için avantaj sağlanmaktadır.

2.3.2. Konumlandırma Modelleri

Bulut bilişim modellerini kullanım şekillerine göre gruplandırmak mümkündür. Bulut modelleri konumlarına göre kamusal, özel ve melez bulut modeli olarak 3 temel gruba ayrılmaktadır. Her modelin sağladığı avantajlar olduğu gibi barındırdığı dezavantajlar da bulunmaktadır. Bulut servisinden yararlanacak kullanıcılar ekonomi, güvenlik ve yönetim önceliklerine uygun olarak konumlandırma modeline karar vermelidir.

Kamusal bulut herkesin kullanımına açık olan ve bir hizmet olarak satılan bulut bilişim altyapısını ifade etmektedir. Bu bulut modelinde bulut bilişim kaynakları diğer kullanıcılar ile ortaklaşa kullanılmaktadır. Bütün bir altyapıyı kiralamak yerine sadece kullanılan kaynak kadar ödeme yapılması sebebiyle bu model ekonomik avantaj sağlamaktadır. Fakat ortak kullanımdan doğan nedenlerle güvenlik önlemlerinin alınmasında kullanıcıların tam yetkisi bulunmamaktadır.

Özel bulut sadece belirli bir organizasyon için sağlanan bulut bilişim altyapısını ifade etmektedir. Özel bulut üzerinden sağlanan kaynaklara sadece kullanıcının bulunduğu organizasyonda çalışanlar erişebilmektedir. Genellikle güvenlik duvarı kullanılarak sadece organizasyonun bulunduğu ağdan gelen şifrelenmiş talepler kabul edilmektedir. Özel bulut modellerinde kamusal modellere göre daha fazla güvenlik önlemi alınabilmesi nedeniyle veri gizliliğinin yüksek önem teşkil ettiği uygulamalarda özel bulut modelleri tercih edilmektedir. Fakat güvenlik önlemlerinin kullanıcı tarafından alınması sebebiyle kullanıcıya düşen yönetimsel yük artmaktadır.

Melez bulut, iki ya da daha fazla bulut modelinin birleşimi ile oluşan bulut modelidir. Bir organizasyon için hem kamusal hem de özel bulut altyapısının beraber kullanılması ile iki modelin avantajlarından faydalanmak mümkündür. Yüksek önemli veri ve uygulamalar özel bulut üzerinden servis edilirken, hassasiyet seviyesi düşük olan veriler ile yapılan işlemler kamusal bulut üzerinde gerçekleştirilebilmektedir. Bu yöntemle özel bulut modelinin güvenlik avantajları ile kamusal modelin ekonomik avantajlarından yararlanılmaktadır. Ayrıca

organizasyon dıřında yer alan kiřilere eriřim verilmesi melez model ile m¼mk¼n olmaktadır. Melez bulutların bir eřidi olan topluluk bulutları ortak ıkarları olan birden fazla organizasyon iin oluřturulmuř bulut biliřim altyapısını ifade etmektedir. Bu modelde organizasyonlar i alıřmalarında ¼zel bulut modelini kullanırken paylařımlarını topluluęa ait ortak bulut servisi ¼zerinden yapabilmektedir.



3. MALZEME VE YÖNTEM

Tez kapsamında tasarladığımız sistemin analiz ve çalıştırma aşamalarında kullandığımız malzeme ve yöntemler bu bölümde açıklanmaktadır. Veri setleri, makine öğrenimi yazılımı kütüphaneleri, hesaplama sırasında kullanılan donanım ve yazılımlar bu tezin malzemelerini oluşturmaktadır. Verinin hazırlanması ve analiz edilmesinde gerçekleştirilen işlemler, makine öğrenimi algoritmaları ve sistem tasarımları ise kullanılan yöntemlere örnek verilebilir.

Veri Setleri bölümünde veri analizinde kullandığımız veri setleri sunulmaktadır. Literatürde yer alan bir veri setinde bulunan veriler ve kendi oluşturduğumuz veri setinin hazırlanışı hakkında detaylı bilgi bu bölümde verilmektedir. Bu bölümde sunulan malzemeler ile makine öğrenimini modelinin oluşturulmasında ve başarısının test edilemesinde kullanılmaktadır. *Makine Öğrenimi* bölümünde kullanıcılardan gelecek verilere uygun olarak sonuçlar üretecek makine öğrenimi modelinin eğitilmesi ve test edilmesinde kullanılan yöntemler aktarılmaktadır. Modelin işleyişi ve çeşitli yazılımlar kullanılarak oluşturulması hakkında bilgi verilmektedir. *Mobil Yazılım* bölümünde sistemin önemli bir parçası olan akıllı telefon üzerinde çalışacak yazılımın özellikleri sunulmaktadır. Mobil arayüz tasarımının yanı sıra kullanıcı verilerine erişimde ve makine öğreniminde mobil ortamda uygulanan yöntemler bu bölümde açıklanmaktadır. *Bulut Yazılımı* bölümünde kullanıcı ve veri yönetimi yöntemleri ile sunucu üzerinde gerçekleştirilen makine öğrenimi yöntemleri aktarılmaktadır. Ayrıca internet sayfası arayüzü tasarımı hakkında bilgi verilmektedir.

3.1. VERİ SETLERİ

Akademik çalışmaların herkes tarafından tekrar edilebilir olması yapılan çalışmanın gücünü arttıran bir özelliktir. Bu kapsamda literatürde bulunan bazı çalışmalar sadece yöntem önerisinde bulunmayıp aynı zamanda kullandıkları veri setlerini de paylaşarak çalışmalarının tekrarlanabilir olduğunu ispatlamaktadır. Ayrıca paylaşılan ortak bir veri seti üzerinde çalışmak, sunulan yöntemlerin de geliştirilmesine olanak tanımaktadır.

Literatürde sağlık verileri üzerine yapılmış olan çalışmaların bir kısmı fiziksel aktivitelere ait veriler barındırmaktadır. *Makine Öğrenimi* bölümünde sunulan yöntemlerin değerlendirilmesi

için öncelikle literatürde paylaşılan bir veri seti üzerinde çalışma gerçekleştirilmiştir. *Mevcut Veri Seti* bölümünde bu veri seti hakkında detaylı bilgi verilmektedir. Ayrıca sunulan yöntemler kendi oluşturduğumuz veri seti üzerinde çalıştırılarak benzer sonuçların elde edebileceğini test edilmiştir. *Oluşturulan Veri Seti* bölümünde bu amaç için oluşturulan veri seti ile ilgili bilgiler sunulmaktadır.

3.1.1. Mevcut Veri Seti

StudentLife isimli veri seti Dartmouth Üniversitesi'nde bulunan 49 adet öğrenciden 10 hafta boyunca toplanmış birçok veriyi barındırmaktadır [11]. Bu zengin veri seti öğrencilerin eğitimleri ile bilgilerden, çoktan seçmeli anket sorularına verilen yanıtlardan ve çeşitli algılayıcılardan toplanan verilerden oluşmaktadır.

Eğitim bilgileri öğrencinin o dönem hangi dersleri aldığı, derslerde verilen ödevlerin günlük olarak sayısı, öğrencinin not ortalaması ve elektronik öğrenme platformu Piazza'yı kullanım istatistiklerini sunmaktadır. Anket yöntemiyle toplanan bilgiler ise öğrencilere verilen zihinsel sağlık ile ilgili bir ankettan veri toplamaya başlamadan önce ve veri toplama işlemi bittikten sonra alınan yanıtlardan oluşmaktadır. Ayrıca 10 haftalık süre boyunca öğrencilere günlük yaşantıları ile ilgili soruların sorulduğu bir anket çalışması yapılmıştır. Öğrenciler kendilerine verilen akıllı telefonlar üzerinden günlük olarak istedikleri sorulara yanıt vererek daha fazla bilgi toplanmasına katkı sağlamaktadır. Bunun yanı sıra akıllı telefonlardaki algılayıcılar kullanılarak 10 farklı algılayıcı verisi toplanmıştır. Bu veriler fiziksel aktivite düzeyi, ortamdaki gürültü düzeyi, konuşma zamanları, karanlıkta geçirilen zamanlar, telefon pilinin doldurulduğu zamanlar, telefonun kilitli olduğu zamanlar, GPS, civardaki Bluetooth cihazlar, civardaki kablosuz erişim noktaları ve bunların kampüsteki konumu ile ilgili bilgileri içermektedir. Kişisel bilgilerin korunması adına veriler paylaşılmadan önce anonimleştirme işlemleri uygulanmıştır.

StudentLife veri seti içerisindeki algılayıcılar ile toplanmış veriler arasında bulunan fiziksel aktivite verileri bu tez kapsamında oluşturduğumuz modelin test edilmesi için uygundur. Yaptıkları çalışmada ivmeölçerler ile 2 saniyede bir gerçekleştirilen ölçümler sonucunda kullanıcıların fiziksel aktivite durumları saptanmıştır. Telefonların batarya sürelerinin uzatılması amacıyla, geliştirilen uygulamada 1 dakika boyunca ivmeölçerlerden aktif ölçüm yapıp 3 dakika uyku süresi tanımlanmıştır. Yapılan her ölçüme ait fiziksel aktivite durum bilgisi zaman etiketiyle birlikte kaydedilmiştir. Durum bilgisinin numerik olarak (Durağan: 0,

Tablo 3.1: StudentLife veri setinden fiziksel aktivite verisi örneği.

Zaman Etiketi	Aktivite Durumu
1364383644	0 (Durağan)
1364383649	1 (Yürüme)
1364383652	0 (Durağan)
1364383654	1 (Yürüme)
1364383657	1 (Yürüme)

Yürüme: 1, Koşu: 2, Belirsiz: 3) tanımlandığı StudentLife veri setinden alınan fiziksel aktivite verisi örneği Tablo 3.1 üzerinde gösterilmektedir.

Zaman etiketi bilgisi UNIX Zamanı formatında verilmektedir. Bu formattaki zaman bilgisi 1 Ocak 1970 günü saat 00:00'dan itibaren geçen saniye sayısını ifade etmektedir [12]. Örneğin 1364383644 zaman etiketi ele alındığında ilgili zamanın Batı Avrupa Zaman Dilimi olan UTC+0'a göre 27 Mart 2013 günü saat 11:27'yi 24 saniye geçtiğini ifade ettiği anlaşılmaktadır.

Oluşturduğumuz modelde fiziksel olarak hareketsiz olma oranı bilgisi girdi olarak kullanılmaktadır. Durağan olma oranı olarak tanımlanan bu değişken, veri setinde sunulan aktivite durumu bilgilerinden yararlanılarak hesaplanmaktadır. Belirlenmiş zaman aralığı içerisinde kişinin fiziksel aktivite açısından hareketsiz kaldığı zamanın belirlenen zaman aralığının tamamına oranı durağan olma oranını vermektedir. He ve Agu tarafından yapılan bir çalışmada tanımlanan bu oran oto-regresyon modeli ile kullanılarak gelecek zaman periyodunda kişilerin hareketsiz olma oranı hesaplanmaktadır [7]. Fakat onların yaptığı hesaplamadan farklı olarak bizim yaptığımız hesaplamada kişilerden aktivite durum bilgisinin alınmadığı zamanlar için kişilerin durağan olduğu varsayılmaktadır.

Hesaplama ile ilgili örnek vermek gerekirse, belirlenmiş zaman dilimi içerisinde 1800 adet ölçüm verisi bulunan bir kişi için 1200 adet veri durağan olarak etiketli ise kişinin durağan olma oranı 0.67 olarak hesaplanmaktadır. Eğer belirlenmiş zaman dilimi için hiç ölçüm verisi yoksa durağan olma oranının 1 olduğu kabul edilmektedir. Durağan olma oranının hesaplanması 3.1 numaralı formülde gösterilmektedir. Fiziksel aktivite açısından durağan olma oranının O sembolü ile gösterildiği formülde Durağan etiketine sahip veri sayısı D , Yürüme etiketine sahip veri sayısı Y , Koşu etiketine sahip veri sayısı K , Belirsiz olarak etiketli veri sayısı ise B simgesi ile ifade edilmektedir.

$$O = \frac{D}{D+Y+K+B} \quad (3.1)$$

3.1.2. Oluşturulan Veri Seti

Kişilerin gelecekteki sağlık durumlarının tahminin yapılması için kullanılan makine öğrenimi modelinin sadece bir veri seti üzerinde test edilmesinin yeterli olmayacağını düşünülerek tez kapsamında alternatif bir veri seti oluşturulmuştur. Sınırlı süre içerisinde 3 kişiden toplanan veriler ile bir veri seti oluşturulmuştur.

Kişilerden toplanan veriler hem akıllı bilekliklerde hem de telefonlarda yer alan algılayıcılardan toplanmıştır. Akıllı bileklik olarak Xiaomi şirketi tarafından üretilen Mi Band 2 marka akıllı bileklikler tercih edilmiştir. Cihazın teknik özellikleri Tablo 3.2 üzerinde gösterilmektedir [13]. Akıllı bileklikte yer alan düşük enerji tüketimli ivmeölçer yardımıyla kişilerin atıkları adım sayısı ve katettikleri mesafe hesaplanmaktadır. Ayrıca gece uykusu sırasında yapılan hareketlere göre uyku kalitesinin takip edilebilmesi mümkündür. Oluşturduğumuz veri setinde kullanıcılardan sadece adım sayısı verisi toplanmıştır. Bu veri birer saatlik periyotlarda kaç adım atıldığı bilgisini ve zaman etiketini barındırmaktadır. Mevcut veri seti ile uyumlu olması açısından zaman etiketi bilgisi UNIX Zamanı formatına çevrilmektedir.

Veri toplanması amacıyla akıllı bileklikler ile birlikte Apple şirketine ait iPhone marka telefonlar kullanılmıştır. Akıllı bilekliklerden telefonlara aktarılan veriler Health Kit kütüphanesi içerisindeki veriler ile birleştirilmektedir [14]. Sadece iPhone telefonlarda bulunan bu kütüphanede saklanabilen veri tipleri *Sağlık Verileri* bölümünde açıklanmıştır. Akıllı bilekliklerde olduğu gibi akıllı telefonlardan da sadece adım sayısı verisi toplanmıştır. Verilerin hem akıllı bileklikler hem de telefonlarda yer alan algılayıcılardan toplanması sayesinde fiziksel aktivite sırasında bileklik ya da telefonda birinin kullanıcının yanında bulunmadığı durumda diğer cihazdan alınan veriler tamamlayıcı olarak çalışmaktadır. Toplanan veriler 1 saatlik periyotlarda kaç adım atıldığı bilgisini ve zaman etiketini içermektedir. Verilerde yer alan zaman etiketleri sayesinde aynı anda hem bileklik hem de telefonun taşındığı durumlarda verinin iki defa işlenmesi engellenmektedir. Bu durum Health Kit kütüphanesi tarafında otomatik olarak halledilmektedir.

Tablo 3.2: Xiaomi Mi Band 2 cihazına ait teknik özellikler.

Marka	Xiaomi
Model	Mi Band 2
Algılayıcılar	İvmeölçer, Kalp Atış Sensörü
Ekran	0.42 inç OLED
Bağlanabilirlik	Bluetooth 4.0 BLE
Uyumluluk	Android 5.0, iOS 7
Su Geçirmezlik	IP67
Batarya Tipi	Lityum Polimer
Batarya Kapasitesi	70 mAh
Bekleme Süresi	20 gün
Giriş Akımı	45 mA (standart), 65 mA (maksimum)
Giriş Voltajı	5V
Cihazın Ağırlığı	7 gram
Cihazdaki Malzemeler	Plastik, Alüminyum
Cihazın Boyutları	15.7 mm x 40.3 mm x 10.5 mm (en, boy, yükseklik)
Bileklerdeki Malzemeler	Termoplastik, Alüminyum
Bilekliğin Uzunluğu	235 mm

Elde edilen bu verilerden oluşan veri setinin StudentLife veri setine benzetilmesi amacıyla kişilere ait veriler 10 haftalık parçalara bölünmüştür. Bu yöntemle 3 kişiden 18 adet 10 haftalık veri kümesi elde edilmiştir. Kişinin belirlenen zaman dilimi içerisinde attığı adım sayısı bilgisi fiziksel aktivite açısından durağan olma oranına çevirilmekte ve veri kümeleri oluşturulmaktadır. Şekil 3.1’de veri setinin oluşturulması sırasında yapılan işlemler gösterilmektedir. Adım sayısı verisinden yararlanılarak fiziksel aktivite açısından durağan olma oranı bilgisi hesaplanmaktadır. Sarwar ve çalışma arkadaşları tarafından yapılan bir çalışmada kişinin 1 dakikada atacağı adım sayısı 100 olarak tanımlanmaktadır [3]. Örneğin 10 dakika içerisinde 800 adım atan bir kişinin yürüyüş süresi 8 dakika olmaktadır. Yürüyüş dışında geçirilen süre tüm zaman aralığına oranlandığında durağan olma oranı 0.2 olarak elde edilmektedir.

3.2. MAKİNE ÖĞRENİMİ

3.2.1. Modelin Yapısı

Makine öğrenimi bilgisayarların öğrenerek işlem yapmasını sağlamaktadır. Bu sayede bilgisayarın ne yapması gerektiği açık bir şekilde programlanmadan istenilen işlemler yapılabilmektedir. Makine öğreniminde sıklıkla kullanılan modellerden biri olan yapay sinir

```

girdi      : Bir kişiye ait  $n$  adet veriden oluşan steps veri seti
çıktı     : Belirlenen zaman dilimleri için kişinin fiziksel aktivite açısından durağan olma
              oranını ve saat bilgisini içeren veri kümeleri
parametre:  $p$  bir veri noktasının zaman dilimi uzunluğunu,  $g$  veri kümelerinin eleman sayısını
              belirler

maksimumAdım  $\leftarrow 60 \times 100$ ; // Dakikada en fazla 100 adım atılabilir
for  $i \leftarrow 1$  to  $n$  do
    | oran[ $i$ ]  $\leftarrow 1 - [\text{Minimum}(\text{steps}[i].\text{step}, \text{maksimumAdım}) \div \text{maksimumAdım}]$ ;
    | saat[ $i$ ]  $\leftarrow \text{SaatiHesapla}(\text{steps}[i].\text{time})$ ; // Zaman etiketinden saat hesaplanır
end

/* Veriler  $p$  satırdan oluşan bir matris dizilir */
oran  $\leftarrow \text{MatrisOluştur}(\text{oran}, p)$ ;
saat  $\leftarrow \text{MatrisOluştur}(\text{saat}, p)$ ;

/* Verilerin birleştirilmesi için sütun bazlı ortalama alınır */
oran  $\leftarrow \text{Ortalama}(\text{oran})$ ;
saat  $\leftarrow \text{saat}[1, :]$ ;

/* Kümeleme için veriler  $g$  satırdan oluşan bir matris dizilir */
oran  $\leftarrow \text{MatrisOluştur}(\text{oran}, g)$ ;
saat  $\leftarrow \text{MatrisOluştur}(\text{saat}, g)$ ;

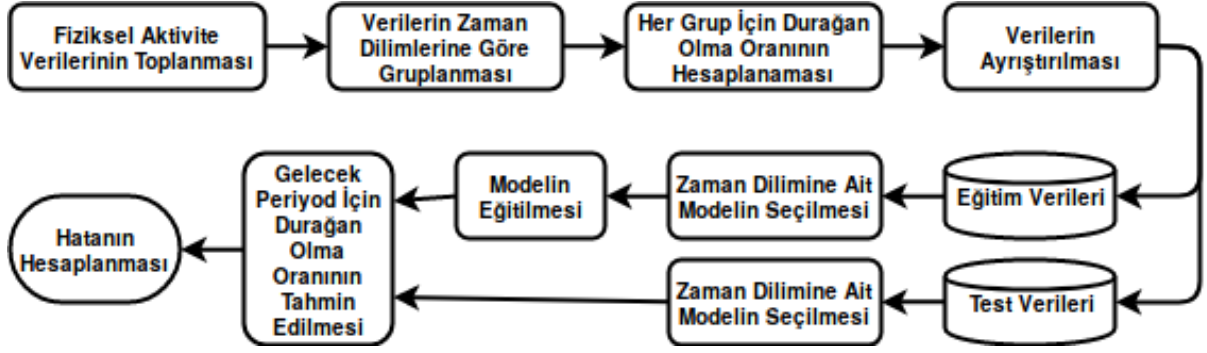
/* Her sütunu ayrı veri kümesi olarak kaydet */
sütunSayısı  $\leftarrow \text{SütunlarıHesapla}(\text{oran})$ ;
for  $j \leftarrow 1$  to sütunSayısı do
    | veriKümesi[ $j$ ].oran  $\leftarrow \text{oran}[:, j]$ ;
    | veriKümesi[ $j$ ].saat  $\leftarrow \text{saat}[:, j]$ ;
end

```

Şekil 3.1: Veri kümelerinin hazırlanmasında kullanılan sözde kod.

ağları hakkında açıklayıcı bilgiler *Yapay Sinir Ağı* bölümünde verilmiştir. Bu bölümde ise yapay sinir ağlarını kullanarak oluşturulan model açıklanmaktadır. Oluşturulan bu modelin kullanıldığı ilk çalışma 28-29 Aralık 2017 tarihlerinde düzenlenen Uluslararası Telekomünikasyon Konferansı'nda sunulmuştur [8].

Makine öğreniminde kullandığımız model oluşturulurken kişilerin gerçekleştirdikleri fiziksel aktiviteler sırasında elde edilen verileri kullanılmıştır. Kullanılan fiziksel aktivite verilerine ait detaylı açıklamalar *Veri Setleri* bölümünde verilmiştir. Modele girdi olarak fiziksel aktivite verilerinden elde edilen durağan olma oranı bilgisi verilmektedir. Modeli oluştururken, kişilerin geçmişte fiziksel aktivitede bulunma oranlarının gelecekte fiziksel aktivitede bulunma oranları ile ilinti olduğu varsayımını yapılmaktadır. Buna göre modelden çıktı olarak önümüzdeki zaman dilimi içerisinde fiziksel aktivitede bulunma oranı bilgisi alınmaktadır.



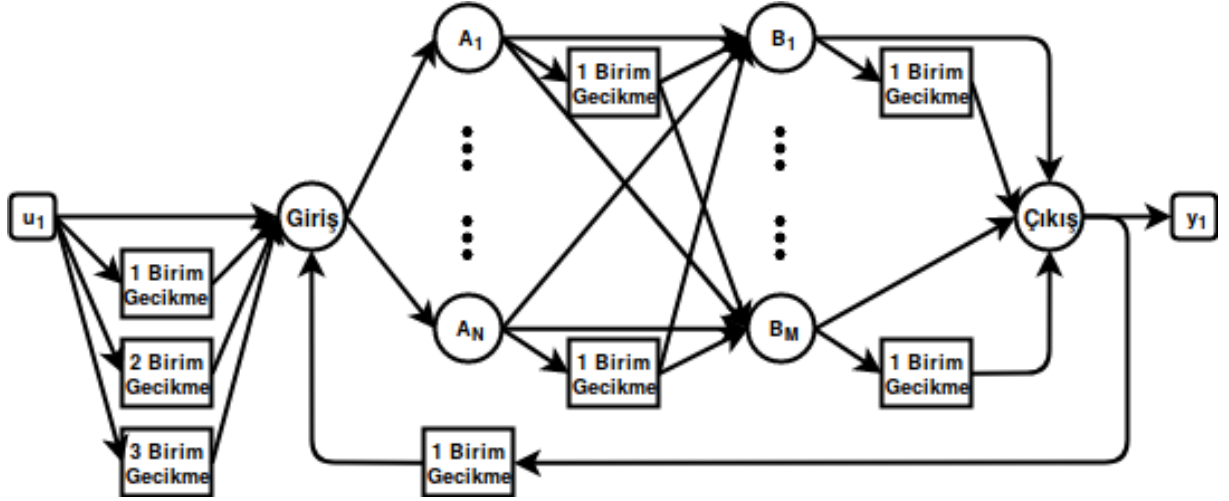
Şekil 3.2: Makine öğreniminde kullandığımız akış diyagramı.

Modelin eğitilmesi ve değerlendirilmesi amacıyla sahip olduğumuz veriler eğitim ve test verisi olarak ikiye ayrılmaktadır. Kişi bazında sahip olunan verilerin yarısı modelin eğitilmesi amacıyla kullanılmaktadır. Kalan veriler ise modelin test edilmesi sırasında kullanılmaktadır. Test verilerinin kullanılması sayesinde oluşturulan modelin performansı daha önce hiç karşılaşmadığı bir veri seti ile test edilmektedir.

Oluşturulan modelde her kişi için tek bir model kullanmak yerine belirli bir periyot içerisindeki zaman dilimi sayısı kadar modelin kullanması tercih edilmiştir. Örnek vermek gerekirse 6 saatlik zaman dilimleri halinde gruplandırılmış verileri kullanarak önümüzdeki 12 saat için 2 adet tahminde bulunmamızın istendiğini ele alalım. Tüm eğitim verisi ile tek bir modeli eğitmek yerine verileri bir günlük periyot içerisindeki zamanlarına göre bölerek dört farklı modeli eğitebiliriz. Bu sayede istenilen 2 adet tahmin için aynı modeli kullanmak yerine tahmin edilmek istenilen zamana göre iki farklı model kullanmış oluruz. Bu tez çalışmasında bu yöntem kullanılarak kişilerin farklı zaman dilimleri içerisinde gerçekleştirdikleri birbirinden farklı davranışları daha etkili bir şekilde modele aktarılmıştır.

Makine öğrenimi için geliştirilen yöntemin genel işleyişi Şekil 3.2’de gösterilmektedir. Modelin çıktısı olarak elde edilen tahmin ile gerçekleşen değer arasındaki fark hesaplanarak modelin performansı ölçülmektedir. Bu performans ölçümünün hesaplanması ve kullanılan modelin performans sonuçları *Makine Öğreniminin Başarısı* bölümünde sunulmuştur.

Aralarında ilişki bulunan verilerin bir zaman serisi şeklinde verildiği durumlarda yinelenen sinir ağı modeli iyi sonuçlar verdiği *Yinelenen Sinir Ağı* bölümünde sunulmuştur. Bu tez kapsamında kullanılacak olan verilerin bu özellikleri taşıdığı varsayılarak modelin oluşturulmasında yinelenen sinir ağı kullanılmıştır. Oluşturduğumuz yinelenen sinir ağı modeli Şekil 3.3’te gösterilmektedir. Modele girdi olarak verilen fiziksel aktivite açısından durağan olma oranı



Şekil 3.3: Yinelenebilir sinir ağı modeli.

Şekil 3.3'te u_1 ile ifade edilmektedir. Veriler 6 saatlik zaman dilimleri halinde gruplanıp zaman dilimlerinin gün içerisinde buldukları konuma göre 4 farklı modele verilmektedir. Bu nedenle bir modele verilen veriler her gün için 1 değerden oluşan bir zaman serisidir. Şekil 3.3'te gösterilen u_1 değişkeni zaman serisindeki ilgili değeri her döngüde modele vermektedir. Şekil 3.3'teki y_1 değişkeni ise modelden çıktı olarak alınan tahmini değeri ifade etmektedir. Yinelenebilir sinir ağlarının yinelenme özelliğinin daha anlaşılır olması amacıyla *gecikme blokları* ile ifade edilmiştir. Yinelenme sayesinde giriş algılayıcısı geçici bir hafızaya sahiptir. Algılayıcıya daha önceden verilen girdiler hafızanın kapasitesine göre daha sonraki döngülerde de kullanılmaktadır. Yinelenebilir sinir ağlarının bu özelliği sayesinde model sadece bir güne ait veri ile beslenirken aynı zamanda önceki 3 güne ait verilerin de kullanılması sağlanmaktadır. Ayrıca modelin çıktısı olarak bir sonraki gün için yapılan tahmini de girdi olarak kullanılarak daha güçlü bir model elde edilmiştir. Şekil 3.3'te A_N ve B_N ile ifade edilen değerler modelde bulunan ara katmanlardaki algılayıcı sayılarını ifade etmektedir. Oluşturulan modelde 2 adet ara katman ve her katmanda 12 adet algılayıcı kullanılmıştır. Yinelenme özelliği giriş katmanının yanı sıra ara katmanlarda da kullanılarak ara katmanlardaki algılayıcıların hafıza sahibi olması sağlanmaktadır.

3.2.2. Modelin Oluşturulması

3.2.2.1. MATLAB ile Model Oluşturma

MATLAB ile veri yapılarının oluşturulması ve sonuçların görsel olarak gösterilmesinde kolaylık sağlaması nedeniyle makine öğrenimi yöntemlerinin uygulanmasında tercih

edilmektedir. Tasarladığımız modelin uygulanması ve performans sonuçlarının elde edilmesinde MATLAB yazılımı kullanılmıştır. MATLAB kullanılarak makine öğrenimi ile ilgili çalışmalar yapabilmek için ücretli ya da ücretsiz olarak dağıtılan paketler bulunmaktadır. Tasarladığımız modelin MATLAB üzerinde uygulanması için ücretsiz ve açık kaynak kodlu olan pyrenn [15] yazılım paketi kullanılmıştır. Bu paket bulunan yinelenen sinir ağı algoritması kullanılarak istenilen sayıda katman ve bu katmanlarda istenildiği kadar algılayıcıyı barındıran ağ modelleri kolaylıkla oluşturulabilmektedir. Bu paket kullanılarak yapılan işlemler Şekil 3.4'te gösterilmektedir. Ayrıca modelin eğitilmesi ve çalıştırılmasına ait akış grafikleri Şekil 3.5 ve Şekil 3.6 üzerinde gösterilmektedir.

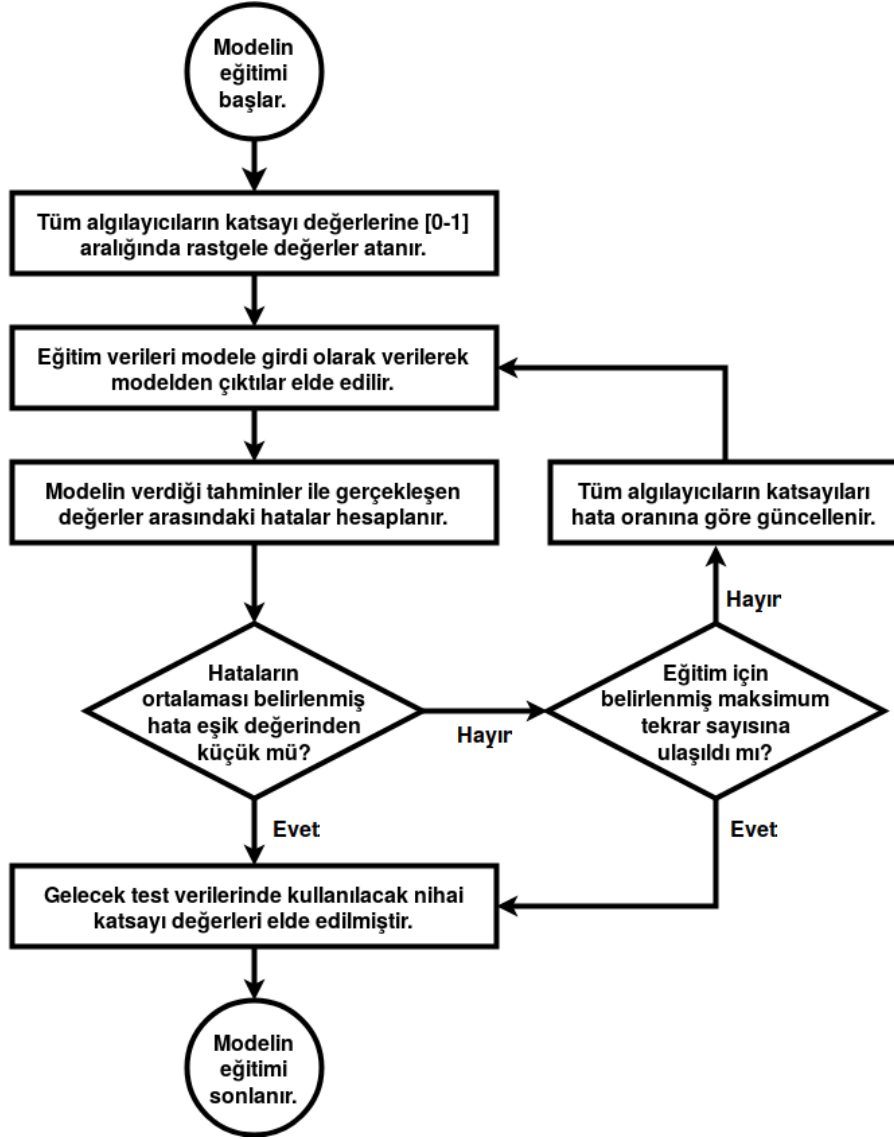
```

girdi      : Her biri  $n$  adet veriden oluşan  $k$  adet veri kümesi dataSets
çikti     : Test verileri için fiziksel aktivite açısından durağan olma oranının tahmini ve veri kümesinin hata oranı
parametre: errorStop ve epochStop eğitim işleminin kesilmesi için yeterli olan hata oranını ve tekrar sayısını, neurons ara katmanlardaki algılayıcı sayısını, inputDelay, hiddenDelay ve outputDelay sırasıyla giriş, ara ve çıkış katmanlarındaki gecikme sayısını belirler

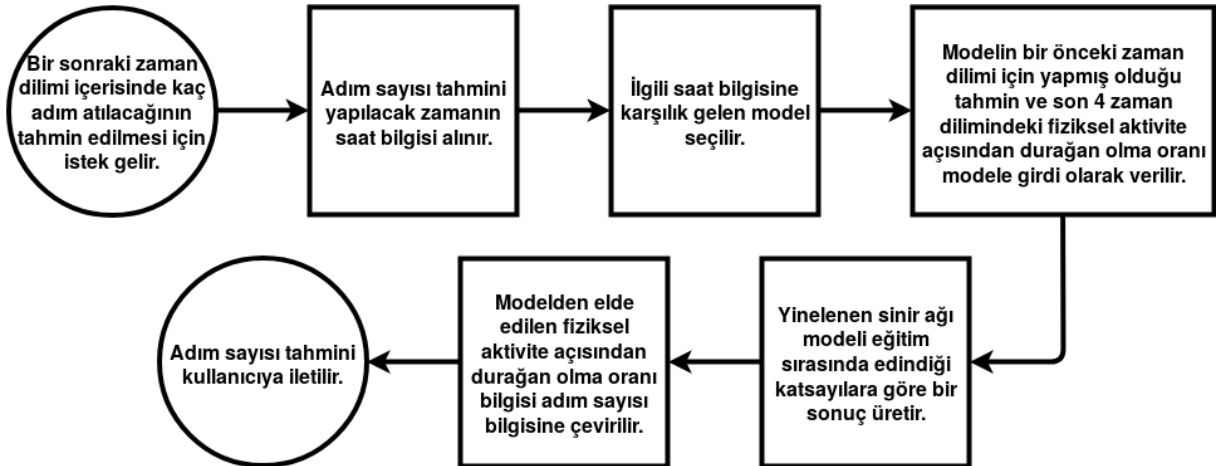
ysa ← YapaySinirAğınıOluştur(neurons,inputDelay,hiddenDelay,outputDelay) ;
for  $i \leftarrow 1$  to  $k$  do
    eğitimVerileri, testVerileri ← VerileriBöl(dataSets[ $i$ ]) ;
    modelSaatleri ← EşsizDeğerleriBul(dataSets[ $i$ ].saat) ;
    foreach  $saat \in$  modelSaatleri do
        modelVerileri ← ModeleAitVerileriBul(eğitimVerileri,  $s_j$ ) ;
        /* Modeller Levenberg - Marquardt ve Broyden - Fletcher - Goldfarb
        - Shanno algoritmaları ile eğitilir */
        modelListesi[ $j$ ].LM ← LMAlgoritmasıylaEğit(modelVerileri, ysa, errorStop,
            epochStop) ;
        modelListesi[ $j$ ].BFGS ← BFGSAlgoritmasıylaEğit(modelVerileri, ysa, errorStop,
            epochStop) ;
    end
    foreach test verisi  $veri_t \in$  testVerileri do
        model ← VeriyəAitModeliBul(modelListesi,  $veri_t$ ) ;
        tahminler[ $i$ ,  $t$ ] ←
            [ModeliÇalıştır(model.LM,  $veri_t$ ) + ModeliÇalıştır(model.BFGS,  $veri_t$ )] ÷ 2 ;
    end
    hataOranı[ $i$ ] ← HatayıHesapla(testVerileri, tahminler) ;
end

```

Şekil 3.4: Modelin eğitilmesi ve test edilmesinde kullanılan sözde kod.



Şekil 3.5: Modelin eğitiminde yapılan işlemlerin akış grafiği.



Şekil 3.6: Adım sayısının tahmininde yapılan işlemlerin akış grafiği.

3.2.2.2. *TensorFlow ile Model Oluřturma*

TensorFlow matematiksel hesaplama amacıyla kullanılan açık kaynak kodlu bir yazılım kütüphanesidir [16]. Android ve iOS işletim sistemleri için yayınlanmış olan yazılım paketleri kullanılarak mobil ortamda makine öğrenimi modelleri çalıştırılmaktadır. Ayrıca Google tarafından sunulan bir bulut bilişim yazılım servisi olan Cloud Machine Learning Engine üzerinde TensorFlow ile oluşturulan modellerin eğitilmesi ve çalıştırılması mümkündür. TensorFlow kütüphanesi içerisinde sunulan yinelenen sinir ağı algoritması kullanılarak bir model oluşturulmuş ve bulut servisi üzerinde çalıştırılmıştır. *Sunucu Üzerinde Makine Öğrenimi* bölümünde modelin kullanımı ile ilgili bilgi verilmektedir.

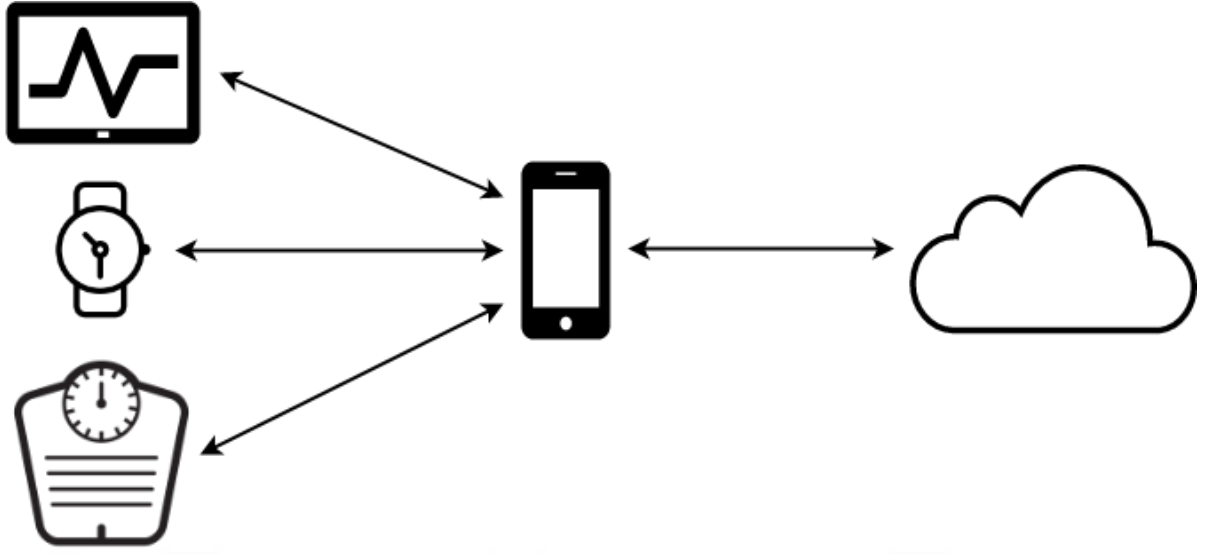
3.2.2.3. *Keras ile Model Oluřturma*

Apple şirketine ait iOS işletim sisteminde bulunan Core ML isimli yazılım kütüphanesi kullanılarak iPhone cihazlarında yüksek performanslarda makine öğrenimi algoritmaları çalıştırılmaktadır. Core ML kütüphanesinin kullanabileceği modeller, desteklenen diğer makine öğrenimi kütüphanelerinin modellerinin dönüřtürülmesi ile elde edilmektedir [17]. Desteklenen kütüphaneler arasında Keras kütüphanesi de bulunmaktadır. Keras yapay sinir ağı için TensorFlow kütüphanesi üzerine geliştirilmiş bir yazılım kütüphanesidir [18]. Bu kütüphane kullanılarak çok daha basit bir şekilde modeller oluşturulabilmektedir. Mobil uygulamada kullanılacak modelin elde edilmesi için ilk olarak Keras ile bir model oluşturulmuştur. Daha sonra Keras modeli Core ML modeline dönüřtürülerek mobil yazılımda kullanılmıştır. *Önceden Eğitilmiş Model* bölümünde modelin kullanımı ile ilgili bilgi verilmektedir.

3.3. MOBİL YAZILIM

3.3.1. Veriye Erişim

Bireylerin sağlık durumları hakkında analiz yapabilmek için ihtiyaç duyulan sağlık verileri normal koşullarda sağlık personelleri tarafından toplanmaktadır. Mobil sağlık uygulamalarının yaygınlaşması ile birlikte sağlık personeli üzerindeki bu yük giderek hafiflemektedir. Bireyler sürekli yanlarında taşıdıkları mobil cihazlardaki algılayıcılar sayesinde sağlık durumları hakkında bilgi edinmektedirler. Cihazlar dahili algılayıcılarından elde ettikleri verileri kendi üzerinde bulunan ekranlarda gösterebilmelerinin yanı sıra farklı cihazlarla haberleşerek bu



Şekil 3.7: Mobil cihazla veriye erişim.

bilgiyi paylaşabilmektedirler. Cihazların kendi aralarında haberleştikleri senaryolarda akıllı cep telefonları genellikle merkezde bulunmaktadır.

Günümüzdeki akıllı telefonlar sahip oldukları kapasite, güç ve bağlanabilirlik özellikleri sayesinde bireysel olarak en çok kullanılan cihaz haline gelmiştir. Kişisel sağlık durumunun takibi ve iyileştirilmesi için geliştirilen mobil telefon uygulamaları hem verinin kullanıcılara sunulmasına olanak sağlamaktadır hem de telefon içerisinde yer alan dahili algılayıcılar sayesinde veriye doğrudan erişim yolu sağlamaktadır. Ayrıca etrafındaki diğer cihazlarla kablosuz iletişim kurarak ihtiyaç duyulan diğer verilerin mobil telefon üzerinde toplanması mümkündür. Toplanan verilerin tümü telefon içerisindeki hafızada saklanabildiği gibi internet erişimi sayesinde uzaktaki bir sunucuya da aktarılabilir. Bu sayede telefonun kaybolması ya da bozulması gibi durumlarda veri kaybının önüne geçilmektedir. Benzer şekilde yeni alınan bir telefon üzerinden eski telefondan sunucuya aktarılan verilere ulaşmak da mümkündür. Bu kapsamda mobil telefonlar yardımıyla sağlık verilerine erişimi dahili verilere erişim, yakındaki cihazlarla ile haberleşme yoluyla erişim ve uzaktaki sunucu ile iletişim kurarak erişim olarak 3 kategoriye ayırabiliriz. Şekil 3.7’de mobil uygulama yardımıyla uzaktaki sunucudan ya da civardaki akıllı tartı, akıllı saat ve farklı tıbbi cihazlardan veriye erişim kurgusu gösterilmektedir.

3.3.1.1. Dahili Algılayıcılar

Akıllı telefonlar içerisinde yer alan algılayıcılardan okunan veriye mobil yazılım üzerinden direkt olarak ulaşmak mümkündür. Dahili olarak bulunan algılayıcılara örnek olarak ivmeölçer, jiroskop, GPS, ışık şiddeti ölçer, manyetik alan ölçer, yakınlık algılayıcısı ve barometre verilebilir. Donanım katmanında bulunan algılayıcılardan okunan verinin yazılım katmanındaki uygulamalar tarafından erişilmesi için işletim sistemi tarafından sağlanan API kütüphanesi kullanılmaktadır. Bu yazılım kütüphanesinin yarattığı soyutlama sayesinde kullanılan cihaz ve algılayıcıların tipi değiştiğinde uygulama aynı yerden verilere erişmeye devam edeceği için yazılımda bir değişiklik yapmaya gerek kalmamaktadır.

Dahili algılayıcılara API kütüphanesi kullanılarak erişmek mümkün olsa rağmen bu algılayıcıların kullanım yetkisinin alınması için yazılım uygulamasının kullanıcıdan izin alması gerekmektedir. Uygulamanın yüklenmesi sırasında ya da ilk defa çalıştırılması sırasında alınan izinler sayesinde algılayıcılara erişim sağlayacak kütüphaneyi kullanmak mümkün olmaktadır. Söz konusu veriler sağlık verileri olduğunda kural koyucular daha da sıkı davranmaktadır. Apple şirketi tarafından üretilen iPhone marka akıllı telefonlarda iOS adı verilen işletim sistemi kullanılmaktadır. Bu işletim sistemi için geliştirilen uygulamalarda sağlık verilerine erişmek için HealthKit adı verilen yazılım geliştirme paketi kullanılmaktadır [14]. Kullanıcıların sağlık verileri bu paket içerisinde şifrelenmiş olarak saklanmaktadır. Bu paket sayesinde birbirinden bağımsız uygulamalar tarafından toplanan tüm sağlık verileri ortak bir yerde toplanmış olur. Böylece kullanıcının izin verdiği uygulamalar diğer uygulamalardan toplanan verilere de erişebilmektedir. Kötü niyetli uygulamaların kullanıcıları kandırıp kişisel verilerine ulaşmasını engellemek amacıyla kullanıcılara uygulama marketi üzerinden sunulan tüm uygulamalar markette yer almadan önce Apple tarafından sıkı bir teste maruz kalmaktadır.

Mobil telefonlarda çalışan bir diğer popüler işletim sistemi ise Google ve Open Handset Alliance tarafından geliştirilen Android adlı işletim sistemidir. Özgür bir yazılım olduğu için birçok cihaz üreticisi tarafından tercih edilmektedir ve yaygın olarak kullanılmaktadır. Android işletim sisteminde sağlık verilerine Google Fit adı verilen yazılım geliştirme paketi aracılığı ile erişilmektedir. Fakat markete koyulan uygulamalar Apple'da olduğu gibi sıkı bir testten geçmediği için Android uygulama marketinde kötü niyetli uygulamalar da yer almaktadır. Bu nedenle birçok geliştirici sağlık verilerini ortak alanda paylaşmak yerine kendi uygulamalarına özel alanda saklamaktadır.

Oluşturulan Veri Seti bölümünde açıklanan verilerin toplanması için tez kapsamında iOS işletim sisteminde çalışan bir uygulama geliştirilmiştir. Sunulan yazılım kütüphaneleri sayesinde ivmeölçerlerden elde edilen verilere kolaylıkla erişim sağlanmaktadır. Bu kütüphanelerde algılayıcılardan elde edilen veriler ham olarak verilmek yerine işlenerek kullanıma hazır olarak sunulmaktadır. Bu sayede ham ivmeölçer verisi üzerinde sinyal işleme yöntemlerini kullanarak adım sayısını hesaplamaya gerek kalmamaktadır. Kullanıcının uygulamaya izin vermesinin ardından geçmiş zamanlara ait adım sayar verilerine de erişmek mümkün olmaktadır.

3.3.1.2. Cıvardaki Cihazlarla Haberleşme

Akıllı telefonlarda çeşitli algılayıcıların bulunmasına rağmen bu algılayıcılar bazı durumlarda yeterli olmayabilmektedir. Bu durumlarda kablosuz haberleşme yardımıyla yakındaki diğer cihazlardaki algılayıcılardan algılanan veriler akıllı telefonlara aktarılabilir. Çok az miktarda enerji kullanarak haberleşmeyi mümkün kıldığı için özellikle sağlık ve spor cihazlarında BLE adı verilen düşük enerjili Bluetooth teknolojisi kullanılmaktadır. Günümüzde tüm akıllı telefonlarda bulunan Bluetooth modülü ile civardaki cihazlarla BLE bağlantısı kurulabilmektedir.

Etraftaki cihazlarda yer alan veriye ulaşabilmek haberleşme işlemlerini yapacak bir mobil yazılımın geliştirilmesi gerekmektedir. Android ya da iOS işletim sistemleri üzerinde çalışacak bir mobil uygulama eğer Bluetooth modülüne erişecek ise gerekli izinlerin uygulamanın sertifikasında belirtilmesi gerekmektedir. Aksi takdirde işletim sistemi tarafından sağlanan yazılım paketi kütüphaneleri kullanılamamaktadır. Sağlık ve spor cihazı üreten birçok üretici kullanıcıların telefonları üzerinden cihazlarındaki veriye ulaşabilmesi için geliştirdikleri mobil yazılımları uygulama marketlerinde sunmaktadır. Bu durumda cihaz üreticisinin sunduğu mobil uygulama, cihazdan edindiği veriyi telefonun hafızasında ya da uzaktaki bir sunucuda saklamaktadır. Hafızaya alınan veriler Apple Health Kit ya da Google Fit gibi ortak bir alanda saklanıyorsa bu veriler kullanıcının izin verdiği diğer uygulamalar tarafından da kullanılabilir. Bu bilgilere erişimin sağlanabilmesi için kullanıcıdan izin alınması gerekmektedir.

Oluşturulan Veri Seti bölümünde açıklanan verilerin toplanılması sırasında hem kullanıcıların telefonlarındaki algılayıcılardan gelen veri hem de akıllı bileklerdeki algılayıcılardan gelen veri kullanılmıştır. Kullanıcılar Xiaomi tarafından üretilen Mi Band 2 marka akıllı bilekliğe sahiptir. Üreticinin kendi geliştirdiği Mi Fit isimli mobil yazılım bilekli ile telefon arasında



Şekil 3.8: Veri erişimi yetki ekranı.

BLE bağlantısı kurarak verileri telefona aktarmaktadır [19]. Bileklikte yer alan ivmeölçer ve nabız ölçer algılayıcıları kullanılarak kişinin adım sayıları, uyku zamanları ve kalp atış hızı bilgisi hesaplanmaktadır. Elde edilen bu bilgiler Apple Health Kit alanında saklandığı için diğer uygulamaların da kullanımına hazır bulunmaktadır. Bizim geliştirdiğimiz mobil uygulama bu ortak alandaki verileri kullanmaktadır. Şekil 3.8’de sağlık verilerine erişim için kullanıcının izin verdiği ekranın görüntüsü gösterilmektedir.

3.3.1.3. Veritabanı Üzerinden Senkronizasyon

Kişilere sağlık durumları hakkında analiz yapmak ve bilgi sunmak için kullanılacak verilerin kişilerin telefonlarında yer alan dahili algılayıcılardan ya da etraflarındaki cihazlardaki algılayıcılardan edinme zorunluluğu bulunmamaktadır. İnternet üzerinden uzaktaki bir sunucuda yer alan sağlık verileri telefon hafızasına alınarak kullanıcıların bilgilendirilmesi mümkündür. Hastane, sağlık merkezi ya da spor salonunda yapılan çalışmalar sonucunda elde

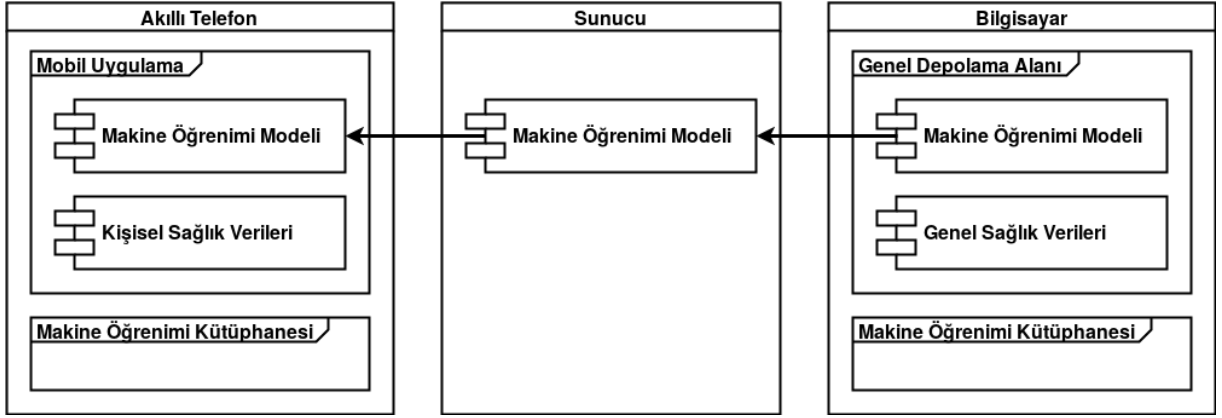
edilen veriler kişiye sunulabilir. Farklı kaynaklardan toplanan veriler analiz edilerek bireyin genel sağlık durumunu takip etmesi sağlanabilir.

Telefondaki ve civardaki cihazlardaki algılayıcılardan toplanan verilerin telefon hafızasında tutulması verilerin kaybolma riskini yaratmaktadır. Bunu engellemek için belirli aralıklarla verilerin bir sunucuya gönderilerek yedeklenmesi veri kaybı riskini azaltacaktır. Telefonun kaybolması ya da bozulması gibi istenmeyen durumlarda ve yeni alınan bir telefona geçişte eski telefondan sunucuya aktarılan verilere ulaşmak mümkün olacaktır. Ayrıca telefon ile sunucu arasında devamlı olarak anlık senkronizasyonun yapılması farklı cihazlardan aynı anda erişim olanağını sağlamaktadır. Bu durumda kişinin kendisi ya da yetki verdiği yakınları ve sağlık personelleri bir internet sayfası aracılığıyla sağlık durumu takibini yapabilecektir. Tez kapsamında geliştirdiğimiz mobil yazılım ile bulut yazılımı senkron çalışmaktadır. Telefondaki ve civardaki cihazlardaki algılayıcılardan toplanan veriler anında sunucudaki veritabanına işlenmektedir. Aynı zamanda kullanıcıların internet sayfası arayüzü üzerinden sunucuya yükledikleri veriler anında telefona aktarılmaktadır. *Bulut Yazılımı* bölümünde bu konu daha detaylı olarak açıklanmaktadır.

3.3.2. Mobil Ortamda Makine Öğrenimi

Bu bölümde *Makine Öğrenimi* bölümünde detaylı açıklamaları verilen modelin mobil ortamda çalıştırılması anlatılmaktadır. Normal koşullarda makine öğrenimi algoritmalarının güçlü bilgisayarlar da çalıştırılması tercih edilmektedir. Fakat verinin yoğun olarak kullanıldığı durumlarda tüm verinin hesaplama işlemi için uzaktaki sunucuya gönderilmesi hem zaman almaktadır hem de sunucuya büyük bir yük getirmektedir. Hızlı tepki vermesi istenilen uygulamalarda sunucuya verilerin gönderilip hesaplama sonucunun gelmesi için yeterli zaman olmayabilmektedir. Bu gibi durumlarda hesaplama işleminin cihaz üzerinde yapılması tercih edilmektedir. TensorFlow yazılım kütüphanesinin Android ve iOS işletim sistemleri için yayınlamış olduğu yazılım paketleri kullanılarak mobil ortamda makine öğrenimi modellerini çalıştırmak mümkün olmaktadır. Ayrıca iOS işletim sisteminde bulunan Core ML isimli yazılım kütüphanesi kullanılarak iPhone cihazlarında daha yüksek performanslarda makine öğrenimi algoritmaları çalıştırılmaktadır.

Makine öğrenimi modellerinin mobil cihazlarda çalıştırılması için iki farklı yöntem tercih edilebilir. *Önceden Eğitilmiş Model* bölümünde açıkladığımız tasarımda kullanıcıya özel



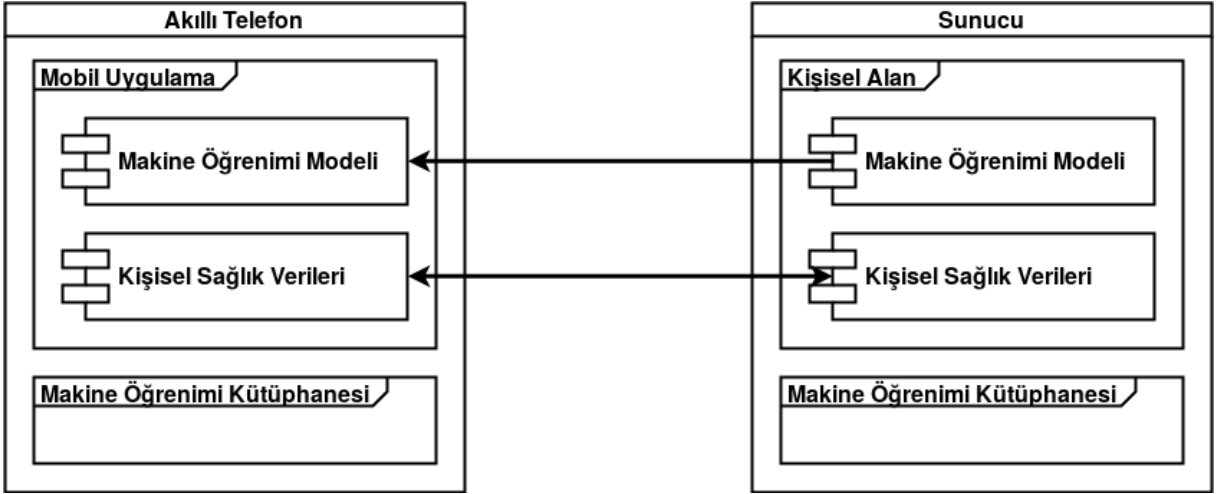
Şekil 3.9: Önceden eğitilmiş model kullanımında sunucu ile haberleşme.

olmayan, geliştirilmiş bir model mobil uygulamayla birlikte sunulmaktadır. *Kullanıcıya Adapte Olan Model* bölümünde ise kullanıcı verilerine göre modelin güncellendiği bir tasarım açıklanmaktadır.

3.3.2.1. Önceden Eğitilmiş Model

İnternet erişiminin olmadığı ya da verilerin telefon dışına gönderilmesinin istenmediği durumlarda makine öğrenimindeki hesaplama işlemleri tamamen mobil ortamda gerçekleştirilmektedir. Bu işlemlerin yapılabilmesi için kullanıcıya özel olmayan geliştirilmiş veriler kullanılarak önceden eğitilmiş bir makine öğrenimi modeli geliştirilen uygulama ile birlikte sunulmaktadır. Kullanıcının kişisel sağlık verileri sadece kendi telefonunda yer alan makine öğrenimi modeline verilerek sonuç hesaplanmaktadır. Bu yöntemde tüm kullanıcılara aynı mobil yazılım paketi sunulacağı için paket içinde verilecek model de aynı olacaktır. Fakat modele girdi olarak verilen sağlık verileri kişiden kişiye farklılık göstereceği için elde edilecek sonuçlar kısmen farklı olacaktır. Şekil 3.9'da önceden eğitilmiş modelin güncellenmesi gösterilmektedir.

Makine öğrenimi modeli eğitilirken sağlık verilerine ve makine öğrenimi kütüphanesine ihtiyaç duyulmaktadır. Bu yöntemde kişisel sağlık verileri eğitim aşamasında kullanılmadığı için eğitim işlemi genel sağlık verileri ile yapılmaktadır. Bu nedenle sunucu üzerinde sağlık verilerinin ve makine öğrenimi kütüphanesinin tutulmasına gerek kalmamaktadır. Eğitim işlemi verilerin ve gerekli kütüphanelerin olduğu bir bilgisayarda yapılabilmektedir. Sadece güncel modelin sunucudan servis edilmesi yeterlidir. Kullanıcılar sunucuda yer alan güncel modeli alarak makine öğrenimi işlemlerini yeni model ile gerçekleştirebilirler. Öte yandan bulut bilişim ile birlikte ihtiyaç duyulan anda istenilen kadar bilişim kaynağı kiralanabilmektedir. Bu



Şekil 3.10: Kullanıcıya adapte olan model kullanımında sunucu ile haberleşme.

sayede daha büyük işlem gücü gerektiren hesaplamaların kısa sürede yapılabilmesi mümkün olmaktadır. Makine öğrenimi modelinin eğitilmesi için bir bilgisayar kullanmak yerine bu işlemlerin bulut bilişim sunucularında yapılması zaman kazandıracaktır.

3.3.2.2. Kullanıcıya Adapte Olan Model

Kullanıcı verilerinin telefon hafızası dışına gönderilmeden kullanılması kişisel sağlık verilerin daha güvende olduğu izlenimini vermektedir. Fakat telefon hafızasındaki verilerin kaybolma ihtimali göz önüne alındığında verileri farklı bir ortamda güvenli olarak saklanması veri güvenliği açısından daha doğru olacaktır. Telefonda toplanan sağlık verileri belirli aralıklarla ya da veri güncellendiğinde anlık olarak uzaktaki bir sunucu gönderilebilmektedir. Kişisel sağlık verileri, diğer kullanıcıların erişemeyeceği kişiye özel ayrılmış güvenli bir alanda saklanmalıdır. Kullanıcı sahibi olduğu verilerin sağlık personeli ya da akrabaları tarafından gözlem amaçlı takip edilmesini istiyorsa gerekli yetkilendirme yapılarak bu verilerin paylaşılması mümkün olmaktadır. *Kullanıcı ve Veri Yönetimi* bölümünde bu durum hakkında detaylı açıklama sunulmaktadır.

Şekil 3.10'da kullanıcıya adapte olan modelin kullanımı gösterilmektedir. Telefon ile sunucu arasında sürekli olarak senkronizasyonun yapılması farklı cihazlardan aynı anda erişim olanağını sağlamaktadır. Bunun yanı sıra kişisel sağlık verilerinin sunucu üzerinde yedeklenmesinin en büyük faydası modelin kullanıcıya adapte olmasıdır. Makine öğrenimi modeli belirlenen periyotlarda kullanıcılardan gelen yeni veriler ile sunucu üzerinde tekrar eğitilebilir. Bu sayede model değişen kullanıcı alışkanlıklarına karşı daha duyarlı olacaktır.

Modelin eğitim işleminin yüksek performanslı bir sunucu üzerinde yapılması sonucu yeni model kısa sürede elde edilmektedir. Kullanıcılar sunucuda yer alan güncel modeli alarak makine öğrenimi işlemlerini yeni model ile gerçekleştirebilirler. Makine öğrenimi hesaplamalarındaki sorgular sunucu yerine telefonda yer alan modele gönderileceği için sunucu üzerinde fazla bir yüklenme olmayacaktır. *Sunucu Üzerinde Makine Öğrenimi* bölümünde modelin eğitilmesi ile ilgili detaylı açıklama verilmektedir.

3.3.3. Mobil Arayüz Tasarımı

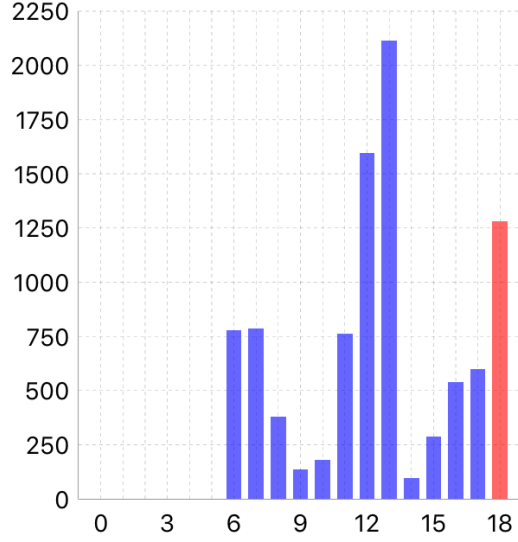
Tez kapsamında kullanıcı verilerinin toplanarak bir veri setinin oluşturulması ve makine öğrenimi modelinin mobil ortamda test edilebilmesi amacıyla iOS işletim sisteminde çalışan bir mobil uygulama geliştirilmiştir. *Civardaki Cihazlarla Haberleşme* bölümünde ifade edilen yöntemler kullanılarak kullanıcıların attıkları adım sayıları ve zamanları toplanmıştır. Bu bilgilere erişimin sağlanabilmesi için kullanıcıdan izin alınması gerekmektedir. Mobil uygulama ile erişilen bu bilgiler bulut üzerindeki veritabanına aktarılarak kullanıcıların uygulama içerisinde oluşturdukları hesapta saklanmaktadır. *Kullanıcı ve Veri Yönetimi* bölümünde konu ile ilgili detaylı açıklamalara yer verilmiştir.

Uygulamanın ana ekranında kullanıcılara gün içerisinde attıkları adım sayıları saatlik bazda bir grafik üzerinde gösterilerek sunulmaktadır. Ayrıca makine öğrenimi yöntemi kullanılarak kullanıcının bir sonraki saat dilimi içerisinde kaç adım atacağı ile ilgili yapılan tahmin gösterilmektedir. Günlük adım sayısının sunulduğu örnek bir ekran Şekil 3.11'de gösterilmektedir. Sunulan grafikte gün içerisinde atılan adımlar mavi ile gelecek saat atılması tahmin edilen adım sayısı kırmızı ile gösterilmektedir. Gelecek saat atılacak adım sayısının tahmini için *Önceden Eğitilmiş Model* bölümünde aktarılan yöntem kullanılmıştır. Fakat *Kullanıcıya Adapte Olan Model ve Modelin Hizmete Sunulması* bölümlerinde sunulan diğer yöntemler kullanılarak aynı sonuçların elde edilmesi mümkündür.

3.4. BULUT YAZILIMI

Kişisel sağlık verilerinin bir sunucu üzerinde saklanması için kullanıcı ve veri yönetimini yapılması için bir yazılıma ihtiyaç duyulmaktadır. Ayrıca makine öğrenimi modelinin sunucu üzerinde eğitilmesi için sunucuda makine öğrenimi kütüphanesinin de çalıştığı bir platforma ihtiyaç vardır. Sunucunun hizmet vermesi için gerekli olan donanımın satın alınması, uygun

Today: 8256, Next Hour: 1279



Back Up

Sign Out

Şekil 3.11: Mobil yazılım ekran görüntüsü.

işletim sistemi ve kütüphanelerin kurulması, gerekli yazılımların satın alınması, güvenlik önlemlerinin alınması ve bakımının yapılması gibi teknik ve yönetim açısından zaman alacak bir dizi süreç bulunmaktadır. *Bulut Bilişim* bölümünde detaylı olarak açıkladığımız nedenler göz önüne alındığında uygun bir servis modeli seçilerek gerekli sistemin oluşturulması zaman ve emek açısından tasarruf sağlayacaktır.

Mevcut ihtiyaçlarımıza baktığımızda Google tarafından sağlanan bulut bilişim servislerinin yeterli olduğu görülmektedir. Google tarafından sunulan Firebase ürünü ile kullanıcı yetkilendirme ve veri yönetimi, App Engine ürünü ile internet uygulamasının servis edilmesi, Cloud Storage ürünü ile veri depolama, Cloud Machine Learning Engine ürünü ile makine öğrenimi modellerinin çalıştırılması, Cloud Shell ürünü ile internet tarayıcısı üzerinden sunucudaki komut satırına erişim yapılabilir.

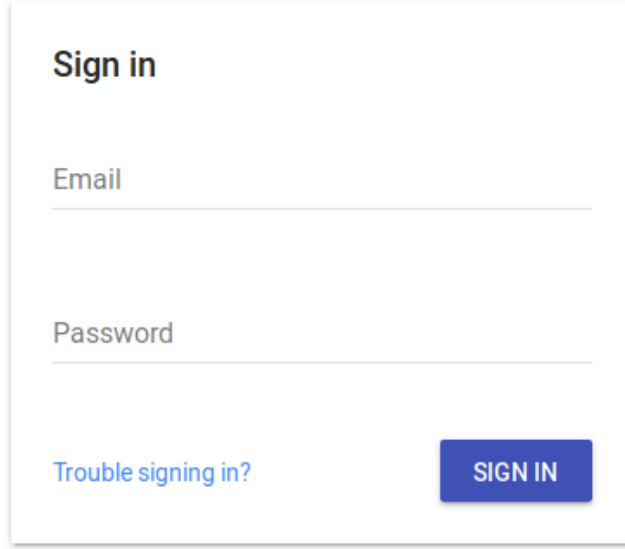
3.4.1. Kullanıcı ve Veri Yönetimi

Sunucu üzerinde yer alan kişisel verilerin güvenliğinin sağlanabilmesi için kullanıcıların uygun olarak yetkilendirilmesi gerekmektedir. Kullanıcıların sahip oldukları kişisel sağlık verilerine sadece kendilerinin ve yetki verdikleri sağlık personeli ve akrabalarının erişim yetkisi olmalıdır. Hangi kullanıcının hangi veriye erişim yetkisinin olduğunun kontrol altında tutulması bir yazılıma ihtiyaç vardır. Google tarafından sunulan bir bulut bilişim yazılım servisi olan Firebase bu ihtiyacımızı karşılamaktadır. Sunulan API kütüphaneleri ile Android ve iOS işletim sistemleri ve internet uygulamaları üzerinden kullanıcı yönetimi yapılabilmektedir. Kişiler e-posta adresi ve şifre bilgilerini girerek kullanıcı hesabı oluşturabilmektedir. Daha sonra mobil ve internet sayfası uygulamaları üzerinden aynı bilgiler ile kullanıcı girişi yapılabilmektedir. Oluşturulan internet uygulamasına e-posta ile giriş sayfası Şekil 3.12’de gösterilmektedir. Ayrıca yetkilendirme yapılırsa Google, Facebook, Twitter ve GitHub sitelerine ait kullanıcı hesapları ya da cep telefonuna gönderilen SMS ile giriş yapılabilmesi mümkündür. Firebase ürünün diğer bir avantajı ise içerisinde yer alan veritabanıdır. Veritabanında saklanan veriler anlık olarak tüm cihazlarda senkron çalışmaktadır. Bu sayede cep telefonundan elde edilen veri anında başka bir cihazdaki internet sayfası uygulamasından görülebilmektedir.

Kullanıcılardan alınan veriler NoSQL veritabanında tutulmaktadır. Verilerin JSON formatında tutulduğu veritabanında ağaç veri yapısı kullanılmaktadır. Veritabanı yöneticisi tarafından girilen kurallara göre veri ağacındaki dallara erişim sağlanmaktadır. Buna göre hangi kullanıcının hangi veriyi okuma ve yazma yetkisine sahip olduğu kural kümesi ile yönetilmektedir. Girilen kurallara uygun olarak kullanıcıların yetkilendirilmesini Firebase servisi sağlamaktadır. Tez kapsamında geliştirilen mobil ve bulut yazılımlarında kullanıcıların sadece kendi verilerine erişimi bulunmaktadır. Fakat gerekli kurallar eklenerek yetki verilen diğer kullanıcılar için erişim sağlanması mümkündür. Bunun için veri sahibi olan kullanıcıdan yetki vermek istediği kullanıcının e-posta adresini girmesi istenecektir. Girilen bilgi var olan bir kullanıcıya karşılık geliyorsa veritabanındaki veri sahibinin yetki verdiği kullanıcılar listesine eklenecektir. Kullanıcı daha sonra yetkilendirme sayfasından yetki verdiği kullanıcıların yetkisini isterse kaldırabilecektir.

Hareketlen Web App

You are signed out.



Sign in

Email

Password

[Trouble signing in?](#) **SIGN IN**

Şekil 3.12: İnternet uygulamasına giriş ekranı.

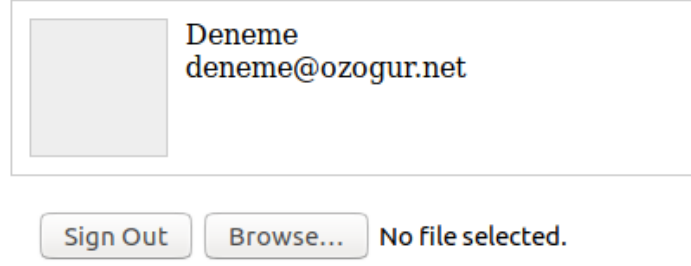
3.4.2. İnternet Sayfası Arayüzü Tasarımı

Bulut üzerinde çalışan veritabanında toplanan kullanıcı verilerine erişimin ve makine öğrenimi modelinin test edilebilmesi amacıyla tez kapsamında sade bir internet uygulaması oluşturulmuştur. Kullanıcılar mobil uygulamayı kullanarak oluşturdukları hesaplarına ait bilgileri internet sayfasındaki giriş ekranına girerek kendi verilerine erişebilmektedirler. Ayrıca internet uygulaması kullanılarak sağlık verilerinin bulunduğu bilgisayardaki bir dosyadan veritabanına veri eklenmesi mümkündür. Kullanıcının günlük attığı adım sayısını görebildiği ekranın görüntüsü Şekil 3.13'te gösterilmektedir. Ek olarak *Modelin Hizmete Sunulması* bölümünde sunulan yöntem test edilmiştir. İnternet uygulaması üzerinden kullanıcının gelecek saatte atacağı adım sayısının tahmini bulut üzerinde çalışan model kullanılarak elde edilmiştir.

3.4.3. Sunucu Üzerinde Makine Öğrenimi

Mobil Ortamda Makine Öğrenimi bölümünde sunucu üzerinde yer alan makine öğrenimi modeline sorgu gönderilmesi ile ilgili çekinceler aktarılmıştır. Verinin yoğun olarak

Hareketlen Web App



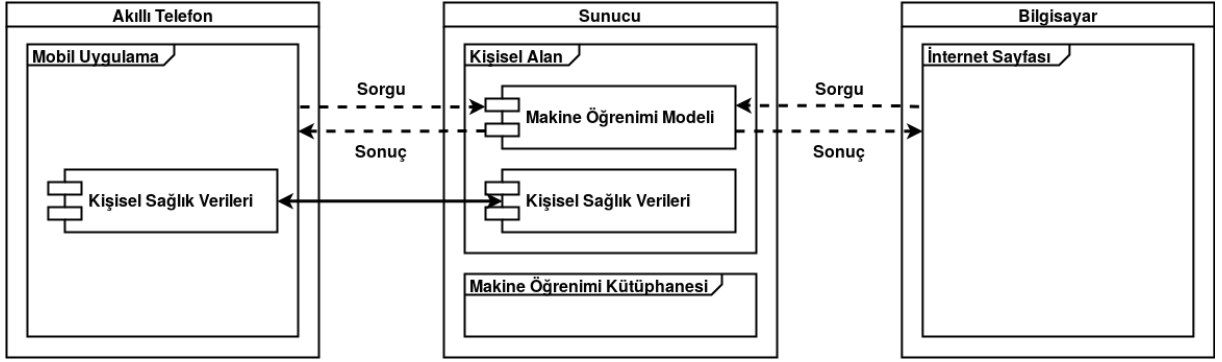
Total steps today: 170

Şekil 3.13: İnternet uygulaması ekran görüntüsü.

kullanıldığı durumlarda tüm sorguların sunucuya gönderilmesi sunucu üzerinde büyük bir yük oluşturmaktadır. Ayrıca bilişim altyapısının kalitesine bağlı olarak sunucudan cevap gelmesi göreceli olarak uzun sürebilmektedir. Anında sonuca ulaşılması beklenen durumlarda sunucudaki makine öğrenimi modelini kullanmak yerine mobil ortamdaki modeli kullanmak daha mantıklı olacaktır. Fakat yoğun veri akışının olmadığı durumlarda makine öğrenimi modelinin sunucu üzerinden hizmet vermesinin faydaları bulunmaktadır. Özellikle makine öğrenimi kütüphanesinin çalıştırılmasının mümkün olmadığı telefonlarda makine öğrenimi modelinin mecburen sunucuda hizmet vermesi gerekmektedir. Aynı durum internet sayfası üzerinden sorgu yapmak için de geçerlidir. İstemci tarafında makine öğrenimi kütüphanesinin kullanılmadığı durumlarda sunucu üzerine istenilen kütüphane kurularak hesaplamalar yapılabilir. Ayrıca modelin sunucu üzerinden hizmet vermesi sayesinde mobil uygulamada makine öğrenimi kütüphanesini barındırmaya gerek kalmayacağı için uygulamanın boyutu da azalacaktır. Makine öğrenimi algoritmasının sadece sunucuda yer alan platformda çalıştırılması sayesinde Android ve iOS gibi farklı platformlara çözüm geliştirme süreci yaşanmayacaktır. Bu sayede mobil yazılım geliştirme ve hata ayıklama süreçleri daha kısa sürecektir. Makine öğrenimi modelinin sunucu üzerinde eğitilmesi ve hizmet vermesi ile ilgili açıklamalar *Modelin Eğitilmesi* ve *Modelin Hizmete Sunulması* bölümlerinde verilmektedir.

3.4.3.1. Modelin Eğitilmesi

Makine öğrenim modelinin eğitilmesi için verilere ve makine öğrenimi kütüphanesinin çalışacağı bir platforma ihtiyaç vardır. Google tarafından sunulan bir bulut bilişim yazılım servisi olan Cloud Machine Learning Engine platform ihtiyacımızı karşılamaktadır. TensorFlow yazılım kütüphanesi ile oluşturulmuş modellerin bulutta çalışmasını sağlayan bu servis verilere



Şekil 3.14: Modelin sunucu üzerinde hizmet sunması.

Google tarafından sunulan bir bulut bilişim veri depolama servisi olan Cloud Storage ürünü üzerinden erişmektedir. Bu veri depolama alanındaki verilere erişim için kullanıcı yönetim sistemi kullanılmaktadır. Bu sayede kullanıcıların verilerinin güvenliği sağlanmaktadır. Kullandığımız tüm bulut bilişim ürünlerinin Google tarafından sağlanması sayesinde kullanıcı yönetimi daha pratik bir şekilde yapılabilmektedir.

Cloud Machine Learning Engine servisini kullanarak modeli eğitmek için makine öğrenimi algoritmasının nasıl çalışacağını anlatan model dosyasının ve kullanılacak verilerin Cloud Storage ürünündeki adreslerine ihtiyaç vardır. Veri ve model dosyalarının adresleri ile birlikte eğitim sonucunda oluşacak modelin depolanacağı adresin sunucuda yer alan komut dosyasında belirtilmesi gerekmektedir. Komut dosyasının çalıştırılmasının ardından satın alınan bilişim kaynakları doğrultusunda işlem yapılmaktadır. Daha fazla kaynak ayrılması durumunda işlem daha kısa sürmektedir. İnternet tarayıcısı üzerinden sunucudaki komut satırına erişim imkanı sağlayan Cloud Shell ürünü kullanılarak komut dosyasının çalıştırılması mümkündür. Bu sayede sunucuya bağlanmak için gerekli yazılım kütüphanelerini kurmadan herhangi bir bilgisayardan modelin eğitilmesi işi başlatılabilmektedir.

3.4.3.2. Modelin Hizmete Sunulması

İstemci tarafında makine öğrenimi kütüphanesinin kullanılmadığı platformlarda hesaplama işlemlerinin makine öğrenimi kütüphanesini bulduran bir sunucu üzerinde yapılması gerekmektedir. Sunucu ile istemciler arasındaki haberleşme Şekil 3.14'te gösterilmektedir. Cloud Machine Learning Engine servisi kullanılarak Cloud Storage ürününde depolanan bir makine öğrenimi modeli hizmete sunulabilmektedir. Hizmet edecek modelin bulutta ya da farklı bir bilgisayarda eğitilmiş olması hizmete sunulması açısından bir fark yaratmamaktadır. Google tarafından sunulan REST API kütüphanesi ile herhangi bir cihazdan makine öğrenimi

modeline sorgu talebi gönderilebilmektedir. Gönderilen talebin geçerli olması için şifrelenmiş anahtara sahip olmak gerekmektedir. Bu anahtara sahip olunmadığı ya da anahtarın yetkisi dışında bir talebin gönderildiği durumlarda sunucu hata mesajı göndermektedir. Yetki sahibi olunan durumda ise sorgu ile birlikte gönderilen veri makine öğrenimi modeline girdi olarak verilmektedir. Makine öğrenimi kütüphanesi kullanılarak yapılan işlemler sonucunda hesaplanan çıktı sorgu sonucu olarak geri gönderilmektedir.



4. BULGULAR

Bu bölümünde tez kapsamında oluşturulan makine öğrenimi modeli ile elde edilen sonuçlar gösterilmekte ve modelin başarısı değerlendirilmektedir. Ayrıca sistem mimarisinde uygulanabilecek alternatif çözüm yöntemlerinin sonuçları karşılaştırılmaktadır.

4.1. MAKİNE ÖĞRENİMİNİN BAŞARISI

4.1.1. Performans Ölçüm Yöntemi

Makine öğrenimi yöntemi ile elde ettiğimiz sonuçların başarısını ölçebilmek amacıyla Ortalama Hata Kare (MSE) metriği kullanılmıştır. MSE metriğinin hesaplanma yöntemi 4.1 numaralı formülde gösterilmektedir. Bu metrikte hesaplanan değer ile gerçekleşen değer arasındaki hatanın karesi alınmakta ve daha sonra bu işlem tüm değerler için tekrarlanıp ortalaması alınarak performans sonucu elde edilmektedir. Makine öğrenimi modelinden alınan tahmini değerler dizisi formüldeki \hat{Y} ifadesi yerine, gerçekleşen değerler ise Y ifadesi yerine koyularak modelin ilgili veri setine ait performans sonucu elde edilmektedir.

$$MSE = \frac{1}{n} \sum_{i=1}^n (\hat{Y}_i - Y_i)^2 \quad (4.1)$$

Makine öğrenimi modelinin performans sonuçlarını karşılaştırabilmek amacıyla Referans Yöntem adını verdiğimiz bir model oluşturduk. Referans model tek bir kurala göre çalışan oldukça basit bir modeldir. Bu model kendisine verilen girdilere bir zaman birimi gecikme ekleyerek çıktı olarak vermektedir. Bu sayede referans modelden bir tahmin yapılması istenildiğinde en son gerçekleşen değer tahmin olarak sunulmaktadır. Makine öğrenimi modelinin başarılı olabilmesi için referans modelden daha başarılı olması beklenmektedir.

Tablo 4.1: Mevcut veri seti ile elde edilen sonuçlar.

Kişi No	Makine Öğrenimi	Referans Yöntem	Kişi No	Makine Öğrenimi	Referans Yöntem
44	0.0005341	0.0014425	17	0.0043847	0.0161870
3	0.0007497	0.0023081	27	0.0052560	0.0035266
10	0.0010663	0.0080118	57	0.0052889	0.0252550
23	0.0011769	0.0028271	47	0.0062682	0.0198770
53	0.0011906	0.0031130	25	0.0064794	0.0106090
50	0.0012732	0.0013073	30	0.0068483	0.0205130
15	0.0013299	0.0022649	22	0.0069009	0.0213920
20	0.0014446	0.0040859	2	0.0072682	0.0113910
39	0.0016788	0.0005074	1	0.0075135	0.0116310
18	0.0017248	0.0054256	36	0.0076587	0.0199630
42	0.0020520	0.0049052	31	0.0077726	0.0234590
8	0.0022438	0.0082676	33	0.0077898	0.0113040
41	0.0025112	0.0032204	54	0.0085623	0.0058597
24	0.0030827	0.0106070	13	0.0092615	0.0265290
16	0.0031870	0.0103780	51	0.0093398	0.0312680
19	0.0031922	0.0073954	34	0.0093719	0.0071903
9	0.0032469	0.0131690	4	0.0103130	0.0346710
5	0.0033159	0.0040073	43	0.0107690	0.0278260
12	0.0033407	0.0126610	0	0.0146910	0.0428380
58	0.0034303	0.0111670	14	0.0167760	0.0152890
56	0.0034418	0.0065651	46	0.0211760	0.0451060
45	0.0034900	0.0100520	52	0.0253040	0.0381690
35	0.0039298	0.0051283	32	0.0340140	0.0359610
49	0.0041666	0.0112450	59	0.0603030	0.0344870
7	0.0042167	0.0085212			

4.1.2. Mevcut Veri Setinin Sonuçları

Makine Öğrenimi bölümünde sunulan yöntemi öncelikle *Mevcut Veri Seti* bölümünde açıklanan veriler kullanılarak test edilmiştir. Kişilere ait veriler 6 saatlik zaman dilimlerine ayrılarak her kişi için ayrı ayrı performans ölçümü yapılmıştır. Veri setinde bulunan 49 kişiden 40'ı için makine öğrenimi yöntemi referans yöntemine göre daha başarılı olmuştur. StudentLife veri seti için makine öğrenimi ve referans modellerine ait MSE metriği ile ölçülen performans sonuçları Tablo 4.1'de gösterilmektedir.

Tablo 4.2: Oluşturulan veri seti ile elde edilen sonuçlar.

Küme No	Makine Öğrenimi	Referans Yöntem
16	0.0010693	0.0061427
3	0.0011270	0.0026566
1	0.0012574	0.0040386
17	0.0014148	0.0038558
4	0.0015346	0.0037784
15	0.0015441	0.0044160
11	0.0016257	0.0052670
12	0.0017489	0.0089139
18	0.0018270	0.0035272
9	0.0018821	0.0093565
8	0.0020781	0.0141560
5	0.0024461	0.0077839
6	0.0026074	0.0171840
14	0.0029818	0.0058374
13	0.0034500	0.0031804
2	0.0038179	0.0098488
10	0.0039042	0.0065188
7	0.0040122	0.0242200

4.1.3. Oluşturulan Veri Setinin Sonuçları

Makine öğrenimi yöntemi *Oluşturulan Veri Seti* bölümünde açıklanan verileri ile test edilerek performans sonuçları elde edilmiştir. 3 kişiden toplanan verilerden oluşturulan 18 adet veri kümesi ayrı ayrı test edilmiştir. Kümelerde yer alan veriler 6 saatlik zaman dilimlerine ayrılarak incelenmiştir. 18 adet kümeden 17 tanesi için makine öğrenimi yöntemi referans yöntemine göre daha başarılı olmuştur. Her küme için makine öğrenimi ve referans modellerine ait MSE metriği ile ölçülen performans sonuçları Tablo 4.2’de gösterilmektedir.

Her veri noktası için yapılan tahmini değer ile birlikte gerçekleşen değer 18 veri kümesi için ekte yer alan *Oluşturulan veri seti için yapılan tahminler* bölümünde ayrı ayrı grafiklerde sunulmaktadır. Tablo 4.2’de gösterildiği üzere veri kümeleri içerisinde en başarılı performans hata oranı en düşük olan 16 numaralı veri kümesinde elde edilmiştir. Bu veri kümesi içerisindeki tüm veri noktaları için yapılan tahmini değer ile birlikte gerçekleşen değer grafiği Şekil 4.1’de gösterilmektedir. Toplanan test verileri kırmızı kesik çizgi ile ifade edilmektedir. Makine öğrenimi ile yapılan tahmini sonuçlar mavi düz çizgi ile işaretlenmiştir. 6 saatlik periyotlar



Şekil 4.1: En düşük hatalı veri kümesi.

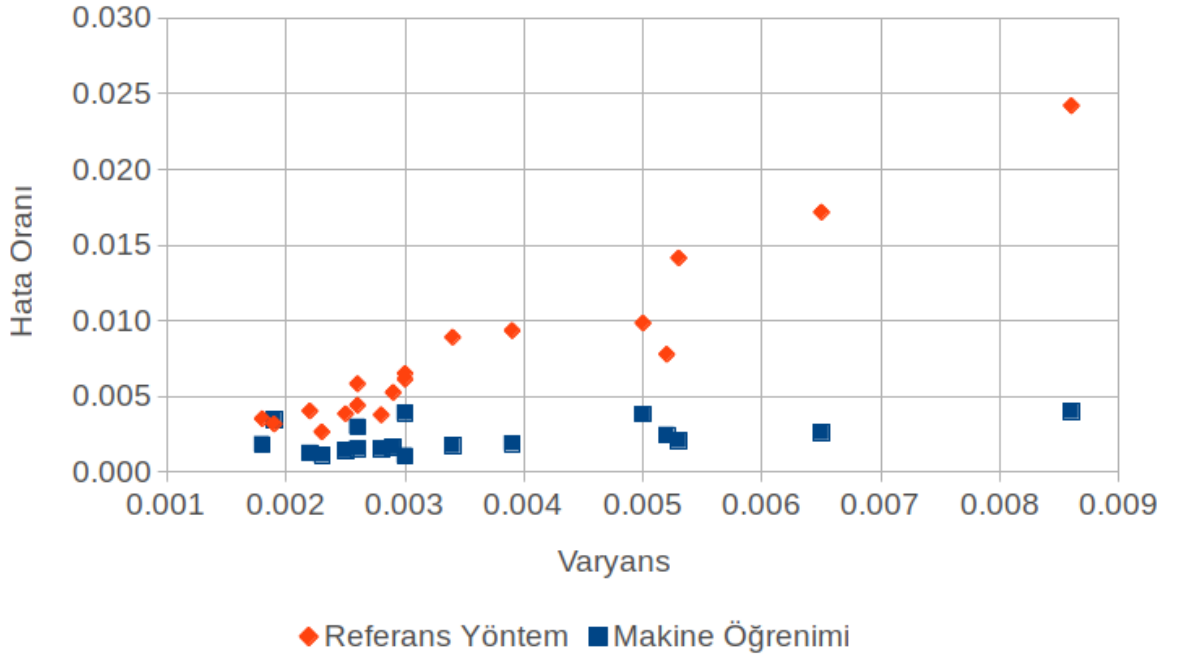


Şekil 4.2: Referans yöntemle kıyasla en başarılı veri kümesi.

ile toplanan sıralı veri yatay ekseninde, kişinin fiziksel aktivite açısından durağan olma yüzdesi dikey ekseninde gösterilmektedir. Referans yöntem ile kıyaslandığında ise en başarılı sonuç 8 numaralı veri kümesi ile elde edilmiştir. Bu veri kümesi için makine öğrenimi ile elde edilen hata oranı, referans yöntem ile elde edilen hata oranının yaklaşık 7’de 1’i kadardır. Bu veri kümesi içerisindeki tüm veri noktaları için yapılan tahmini değer ile birlikte gerçekleşen değer grafiği Şekil 4.2’de gösterilmektedir.

Grafikler incelendiğinde sadece belirli anlarda için tahmin edilen değerlerin gerçekleşen değer ile yakın çıkmasının yeterli olmadığı görülmektedir. Hata oranının düşük olması tahmin edilen değerlerin seri bir şekilde gerçekleşen değerler ile örtüşmesi gerekmektedir. 16 numaralı veri kümesine ait sonuçlara bakıldığında kişinin hareketsiz olma oranı ile tahmin edilen değerlerin çoğunlukla bir seri halinde örtüştüğü görülmektedir. Bu nedenle bu veri kümesine ait sonuçların en düşük hata oranına sahip olması şaşırtıcı değildir.

Tablo 4.2’deki performans sonuçları karşılaştırıldığında makine öğrenimi ile elde edilen sonuçlar arasındaki farkın referans yöntem kadar fazla olmadığı görülebilir. Test verileri incelendiğinde bir veri kümesi içerisindeki verilerin ortalamaya göre dağınık olduğu durumlarda referans yöntemin hata oranının arttığı görülmüştür. Şekil 4.3’te test verilerindeki varyans ile makine öğrenimi ve referans yöntemine ait performans sonuçlarının ilişkisi



Şekil 4.3: Test verisindeki varyansın performans sonuçlarına etkisi.

görülmektedir. Test verisindeki varyansın artışı ile referans yöntemin hata oranının artışı arasında doğru orantı bulunduğu görülmektedir. Buna karşılık makine öğrenimi yönteminin varyanstaki değişimden etkilenmediği anlaşılmaktadır. Bu durum 8 numaralı veri kümesine ait sonuçlarda olduğu gibi neden makine öğrenimi yönteminin referans yöntemine göre daha başarılı olduğunu açıklamaktadır.

4.2. SİSTEM MİMARİSİNİN DEĞERLENDİRİLMESİ

Bu tez çalışmasında, kişilerden akıllı telefonlar aracılığı ile toplanan veriler kullanılarak gelecekteki sağlık durumları hakkında tahmin yapabilecek ve geri bildirim verebilecek bir sistem tasarımının yapılması amaçlanmıştır. Böyle bir sistem için 3 farklı alternatif sunulmuştur. Bu alternatifler mobil ağırlıklı çözüm, bulut ağırlıklı çözüm ve melez çözüm olarak adlandırılabilir. Sunulan çözümlerin karşılaştırılması Tablo 4.3 üzerinde gösterilmektedir.

Mobil ağırlıklı çözüm *Önceden Eğitilmiş Model* bölümünde sunulan çözümü ifade etmektedir. Bu çözümde makine öğreniminin kullanıldığı gelecek sonuçların tahmininin yapıldığı hesaplamalar mobil platformda gerçekleştirilmektedir. Bu sayede tahmin işlemleri hızlı bir şekilde gerçekleştirilmekte ve sunucudan yanıt beklemeye gerek kalmamaktadır. Ayrıca bu çözümde internet erişimi olmadan tahmin yapabilmek de mümkündür. Kullanıcılar sadece

Tablo 4.3: Alternatif çözümlerin özellik tablosu.

Özellikler	Mobil Ağırlıklı Çözüm	Melez Çözüm	Bulut Ağırlıklı Çözüm
Kullanıcıya adapte olabilme	Yok	Var	Var
Veri yedekleme	Yok	Var	Var
Veri paylaşımı	Yok	Var	Var
Farklı cihazlardan durum takibi	Yok	Var	Var
Hasta takibi	Yok	Var	Var
İnternet olmadan tahmin yapabilme	Var	Var	Yok
Tahmin hızı	Yüksek	Yüksek	Ortalama
Sunucu trafiği	Çok düşük	Düşük	Çok yüksek
Mobil makine öğrenimi kütüphanesi gereksinimi	Var	Var	Yok
Bulutta makine öğrenimi kütüphanesi gereksinimi	Yok	Var	Var
Mobil yazılım geliştirme maliyeti	Yüksek	Yüksek	Düşük
Bulut yazılım geliştirme maliyeti	Çok düşük	Yüksek	Yüksek
Bulut işletim maliyeti	Çok düşük	Ortalama	Yüksek

modelde bir güncelleme olması durumunda sunucuyu güncel modeli edinmek amacıyla kullanılmaktadır. Bulut bilişim servisinin sadece depolama alanı olarak kullanılması sebebiyle bulut yazılımı geliştirme ve işletim maliyeti oldukça düşük olacaktır. Diğer taraftan bu çözümde kullanıcıların verileri paylaşılmamaktadır ve sadece akıllı telefonun hafızasında saklanmaktadır. Verilerin paylaşılmaması veri güvenliği açısından olumlu gözükmesine rağmen veri kaybını önlemek amacıyla verilerin yedeklenmesine izin vermemektedir. Kullanıcıların kendi verilerine farklı cihazlardan erişememesine ek olarak kullanıcının yetki verdiği yakınlarının ve sağlık personelinin hasta takibi yapabilmesi de mümkün değildir. Ayrıca kullanıcı verileri modelin eğitilmesinde kullanılmadığı için modelin kullanıcıya adapte olması sağlanamamaktadır. Bu çözümde her mobil platformlar için ayrı uygulama geliştirme ihtiyacı bulunmaktadır. Bu nedenle mobil yazılım geliştirme süresi daha uzun ve maliyetli olacaktır.

Bulut ağırlıklı çözüm *Modelin Hizmete Sunulması* bölümünde sunulan çözümü ifade etmektedir. Bu çözümde makine öğreniminin kullanıldığı gelecek sonuçların tahmininin yapıldığı hesaplamalar bulut üzerinde gerçekleştirilmektedir. Özellikle makine öğrenimi kütüphanesinin kullanılmadığı mobil platformlarda hesaplamaların yapılabilmesi için bu kütüphanenin çalıştığı bir bulut yazılım ihtiyacı bulunmaktadır. Hesaplama işlemlerinin bulut üzerinde yapılması sayesinde sadece sunucuya sorgu gönderip gelen sonucu gösterecek görece daha basit bir mobil yazılımın geliştirilmesi yeterli olacaktır. Bu durumda mobil yazılım

geliştirme maliyeti azalmaktadır fakat bulut yazılım geliştirme maliyeti artmaktadır. Bu çözümde kullanıcıya ait veriler bulut üzerinde güvenli bir şekilde saklanmaktadır. Bu sayede hem kullanıcı farklı cihazlardan sağlık durumunu takip edebilmektedir hem de akıllı telefonun kaybolması ya da bozulması durumunda oluşabilecek veri kaybının önüne geçilmektedir. Ayrıca kullanıcının yetki verdiği yakınlarının ve sağlık personelinin internet uygulaması üzerinden hasta takibi yapabilmesi de mümkündür. Kullanıcı verilerinin bulut üzerinde depolanmasının bir avantajı da modelin kullanıcının davranışındaki değişikliklere adapte olabilmesidir. Mobil ağırlıklı çözümde mümkün olmayan bu özellik sayesinde sağlık durumu ile ilgili yapılan tahminler daha tutarlı olacaktır. Yapılacak her tahmin için sunucuya sorgu gönderilmesi sunucu üzerinde yüksek bir trafik oluşturacaktır ve kullanıcının sunucudan cevap alabilmesi beklediği süre sunucunun yoğunluğuna ve kullanıcının internet hızı ile orantılı olarak değişmektedir. Hem her kullanıcı için farklı modelin eğitilmesi hem de tahmin işlemleri için bulut platformunun kullanılması bulut işletim maliyetini yükseltmektedir.

Mobil ve bulut ağırlıklı çözümlerin olumlu özelliklerinin alındığı melez çözüm *Kullanıcıya Adapte Olan Model* bölümünde sunulan çözümü ifade etmektedir. Bu çözümde gelecek sonuçların tahmini için kullanılacak modelin eğitilmesi bulut üzerinde gerçekleştirilirken modelin çalıştırılması mobil platformda yapılmaktadır. Bu durumda hem mobil hem de bulut platformlarında makine öğrenimi kütüphanelerini kullanılmaktadır. Bunun sonucu olarak mobil ve bulut yazılım geliştirme süresi ve maliyeti artmaktadır. Fakat gelecek sonuçların tahmini için modele gönderilecek sorgular sunucu yerine mobil cihazdaki modele gönderileceği için sunucu üzerindeki trafik hafiflemektedir. Bununla birlikte bulut işletim maliyeti sadece modelin eğitilmesi ve verilerin depolanması için oluşacak ve bulut ağırlıklı çözüme göre daha az olacaktır. Kullanıcı verilerinin bulut üzerinde depolanması sayesinde bulut ağırlıklı çözümde olduğu gibi melez çözümde de veri yedekleme, farklı cihazlardan durum takibi, internet uygulamasıyla hasta takibi mümkün olmaktadır. Bulut üzerinde depolanan kullanıcıya ait veriler belirli aralıklarla kullanılarak makine öğrenimi modeli bulut üzerinde eğitilerek güncellenmektedir. Modelde bir güncelleme olduğunda mobil uygulama bulut üzerinden güncel modeli alarak çalışmaya devam etmektedir. Bu sayede hem model kullanıcıya adapte olmaktadır hem de tahmin için yapılacak işlemler mobil cihaz üzerinde yapıldığı için hızlı bir şekilde sonuçlar elde edilmektedir.

5. TARTIŞMA VE SONUÇ

5.1. GENEL SONUÇ

Tez kapsamında geliştirilen bir mobil uygulama ile hem akıllı telefonlardaki hem de civardaki cihazlarda yer alan algılayıcılardan sağlık verilerinin ne şekilde toplanabileceği gösterilmiştir. Yinelenen sinir ağları kullanılarak oluşturulan model ile kullanıcıların gelecekteki sağlık durumları ile ilgili tahminler yapılmıştır. *Mevcut Veri Setinin Sonuçları* bölümünde sunulduğu üzere 49 kişiden 40'ı için makine öğrenimi yöntemi ile referans yöntemine göre daha başarılı sonuçlar elde edilmiştir. *Oluşturulan Veri Setinin Sonuçları* bölümünde gösterilen sonuçlarda ise 18 adet kümeden 17 tanesi için makine öğrenimi yönteminin daha başarılı olduğu tespit edilmiştir. Sonuç olarak oluşturulan makine öğrenimi yönteminin adım sayısını tahmin etmekte başarılı olduğu görülmektedir.

Sistem mimarisinde uygulanan alternatif çözümler incelendiğinde mobil ağırlıklı çözümde veri paylaşımının olmaması bu çözümü diğer çözümlerin gerisinde bırakmıştır. Bu çözüm sadece hasta takibinin yapılmasına ihtiyaç duyulmayan ya da verilerin mobil cihazdan çıkarılmasına izin verilmeyen durumlarda tercih sebebi olabilir. Diğer taraftan bulut ağırlıklı çözümde veri yedekleme, farklı cihazlardan durum takibi ve internet uygulamasıyla hasta takibi mümkün olmaktadır. Fakat bu çözüm incelendiğinde sunucu üzerinde çok fazla yük olduğu görülmektedir. Her tahmin işlemi için sunucuya sorgu gönderilmesi masraflı olmaktadır. Bu iki çözümün karışımı olarak sunulan melez çözümde ise her iki tarafın avantajlı özellikleri alınmaktadır. Mobil uygulamada bulunan kişisel sağlık verilerinin bulutta yer alan kişisel alanlarda saklanmasıyla verilerin yedeklenmesi yapılmaktadır. Bu sayede kullanıcılar farklı cihazlardan ve internet uygulaması üzerinden sağlık durumlarının takibini yapabilmektedir. Kullanıcıların yanı sıra kullanıcıların yetki verdiği yakınları ve sağlık personelleri de hasta takibini yapabilmektedir. Ayrıca bulut üzerinde yer alan kullanıcı verileri kullanıcıya özel makine öğrenimi modelinin eğitilmesinde kullanılarak kullanıcıya adapte olabilen bir model elde edilmektedir. Bulut üzerinde eğitim işlemleri yapılan bu modelin mobil cihaz üzerinde çalıştırılması sayesinde sunucuya binen yük azaltılmakta ve tahmin sonucuna erişim hızı artmaktadır. Melez çözümün dezavantajlı olan yanı ise hem mobil hem de bulut platformu

için makine öğrenimi kütüphanesine ihtiyaç duymasıdır. Bu gereksinim hem yazılım geliştirme süresini uzatmakta hem de yüm platformlarda desteklenmediği için seçenekleri azaltmaktadır.

5.2. GELECEK ÇALIŞMALAR

Bu çalışmada sunulan makine öğrenimi modeli ile birlikte mobil ve bulut yazılımı tasarımı kullanılarak kişilerin sağlık durumunun takibinin ve gelecek dönemdeki durumlarının tahminin yapılabileceği gösterilmiştir. Gelecekte yapılacak çalışmalarda adım sayısının yanı sıra farklı sağlık parametreleri kullanılabilir. Sunulan modelin ve sistem tasarımının diğer sağlık verileri ile nasıl performans göstereceğinin incelenmesi yapılan bu çalışmayı bir adım ileriye taşıyacaktır.

Hasta takibi özelliği geliştirilerek belirlenen değer aralığının dışına çıkılması durumunda ilgili kişilere uyarı gönderen bir sistem tasarımı yapılabilir. Böyle bir sistemde sağlık durumunun kötüye gittiğinin farkına varmayan hastanın erkenden uyarılması da mümkün olacaktır.

KAYNAKLAR

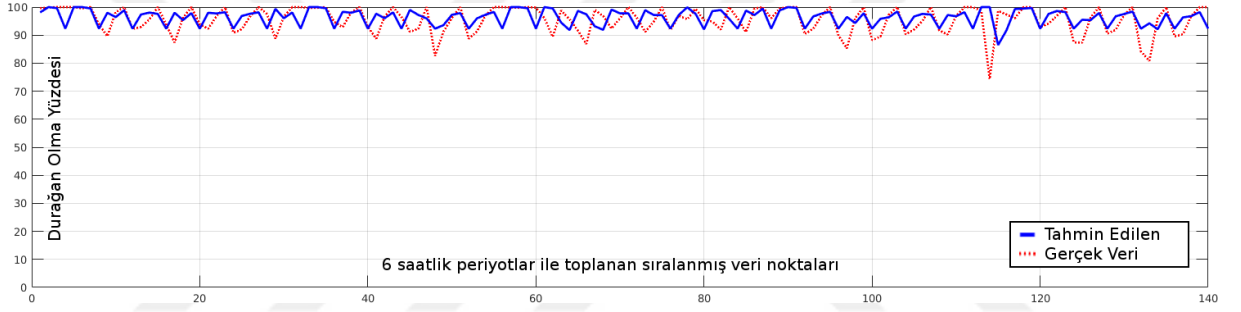
- [1]. A Prestwich, M Perugini, and R Hurling. Can the effects of implementation intentions on exercise be enhanced using text messages? *Psychology and Health*, 24(6):677–687, 2009.
- [2]. Aarti Sathyanarayana, Shafiq Joty, Luis Fernandez-Luque, Ferda Ofli, Jaideep Srivastava, Ahmed Elmagarmid, Teresa Arora, and Shahrad Taheri. Sleep quality prediction from wearable data using deep learning. *JMIR mHealth and uHealth*, 4(4), 2016.
- [3]. Atifa Sarwar, Hamid Mukhtar, Maajid Maqbool, and Djamel Belaid. Smartfit: A step count based mobile application for engagement in physical activities. *International journal of advanced computer science and applications (IJACSA)*, 6(8):271–278, 2015.
- [4]. Sebastian Deterding, Dan Dixon, Rilla Khaled, and Lennart Nacke. From game design elements to gamefulness: defining gamification. In *Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments*, pages 9–15. ACM, 2011.
- [5]. Sunny Consolvo, Predrag Klasnja, David W McDonald, and James A Landay. Goal-setting considerations for persuasive technologies that encourage physical activity. In *Proceedings of the 4th international Conference on Persuasive Technology*, page 8. ACM, 2009.
- [6]. Laura R Pina, Ernesto Ramirez, and William G Griswold. Fitbit+: A behavior-based intervention system to reduce sedentary behavior. In *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2012 6th International Conference on*, pages 175–178. IEEE, 2012.
- [7]. Qian He and Emmanuel O Agu. Towards sedentary lifestyle prevention: An autoregressive model for predicting sedentary behaviors. In *Medical Information and Communication Technology (ISMICT), 2016 10th International Symposium on*, pages 1–5. IEEE, 2016.
- [8]. Gokhan Ozogur, Mehmet Ali Erturk, and Muhammed Ali Aydin. Prediction of physical activity times using deep learning method. In *International Telecommunications Conference: Proceedings of the ITelCon 2017*. Springer, 2018.
- [9]. *Health Data Types | Apple Developer Documentation*. Available at https://developer.apple.com/documentation/healthkit/health_data_types.
- [10]. Peter Mell and Tim Grance. The nist definition of cloud computing. *Communications of the ACM*, 53(6):50, 2010.
- [11]. Rui Wang, Fanglin Chen, Zhenyu Chen, Tianxing Li, Gabriella Harari, Stefanie Tignor, Xia Zhou, Dror Ben-Zeev, and Andrew T Campbell. Studentlife: assessing mental health, academic performance and behavioral trends of college students using smartphones. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 3–14. ACM, 2014.

- [12]. *Unix time* - *Wikipedia*. Available at http://en.wikipedia.org/wiki/Unix_time.
- [13]. *Mi Band 2* - *Mi Global Home*. Available at <https://www.mi.com/en/miband2/>.
- [14]. *Apple HealthKit*. Available at <https://developer.apple.com/healthkit/>.
- [15]. Dennis Atabay. pyrenn: First release, January 2016.
- [16]. Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016.
- [17]. *Converting Trained Models to Core ML* | *Apple Developer Documentation*. Available at https://developer.apple.com/documentation/coreml/converting_trained_models_to_core_ml.
- [18]. François Chollet et al. Keras. <https://keras.io>, 2015.
- [19]. Anhui Huami Information Technology Co. Ltd. Mi fit on the app store, Dec 2014. Available at <https://itunes.apple.com/app/xiao-mi-yun-dong/id938688461>.

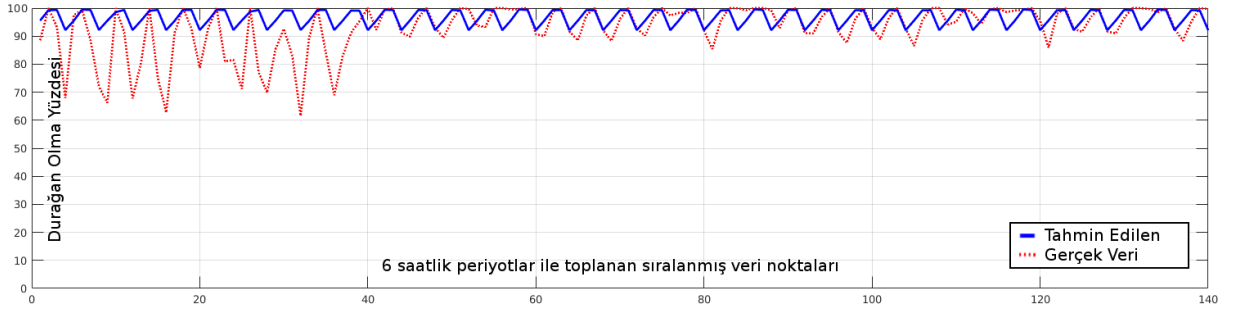
EKLER

EK 1. Oluşturulan veri seti için yapılan tahminler

Bu bölümde, oluşturduğumuz veri seti kullanılarak elde ettiğimiz sonuçlar gösterilmektedir. Kişilerden toplanan test verileri kırmızı kesik çizgi ile ifade edilmektedir. Makine öğrenimi ile yapılan tahmini sonuçlar mavi düz çizgi ile işaretlenmiştir. 6 saatlik periyotlar ile toplanan sıralı veri yatay eksende, kişinin fiziksel aktivite açısından durağan olma yüzdesi dikey eksende gösterilmektedir.



1 numaralı veri seti.

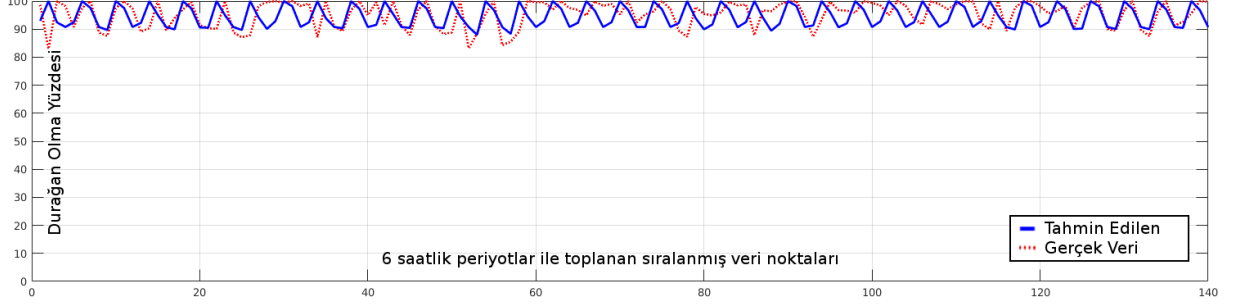


2 numaralı veri seti.

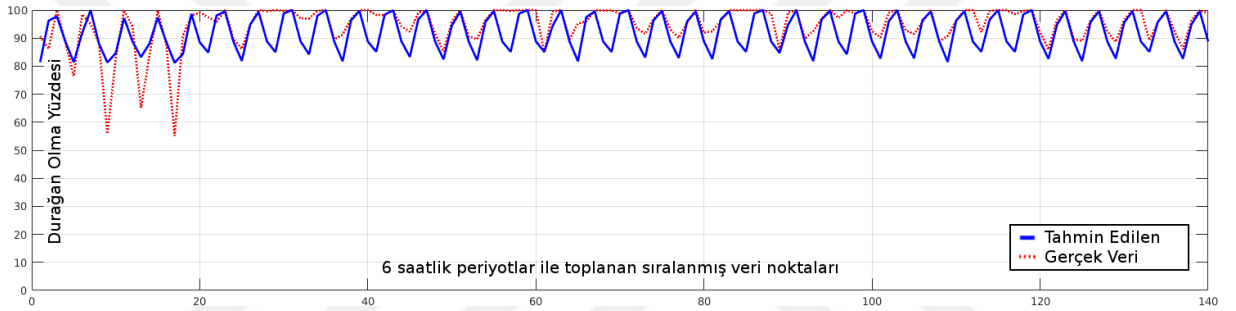


3 numaralı veri seti.

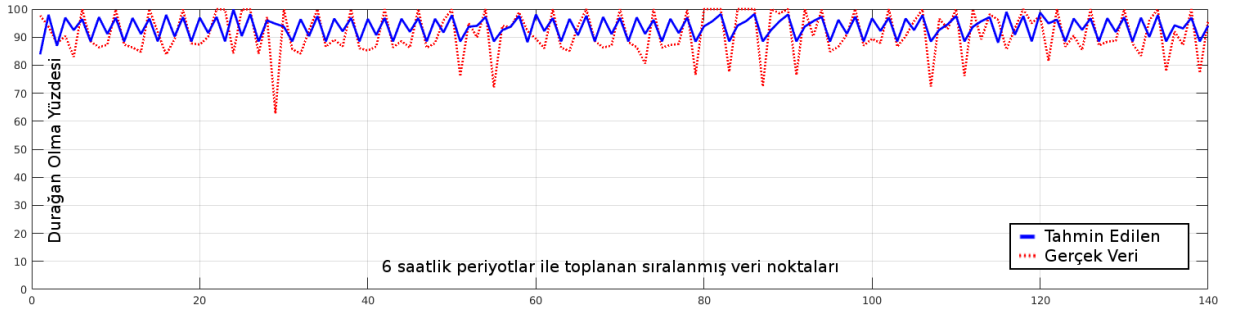
EK 1 (devam):



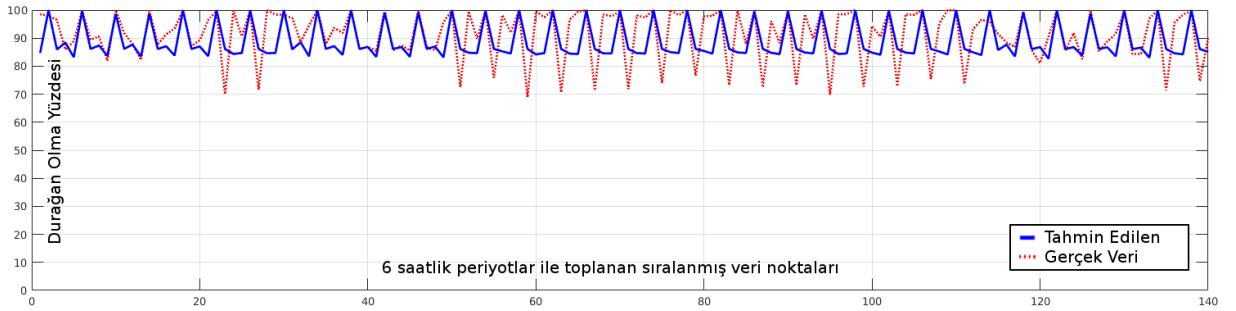
4 numaralı veri seti.



5 numaralı veri seti.



6 numaralı veri seti.

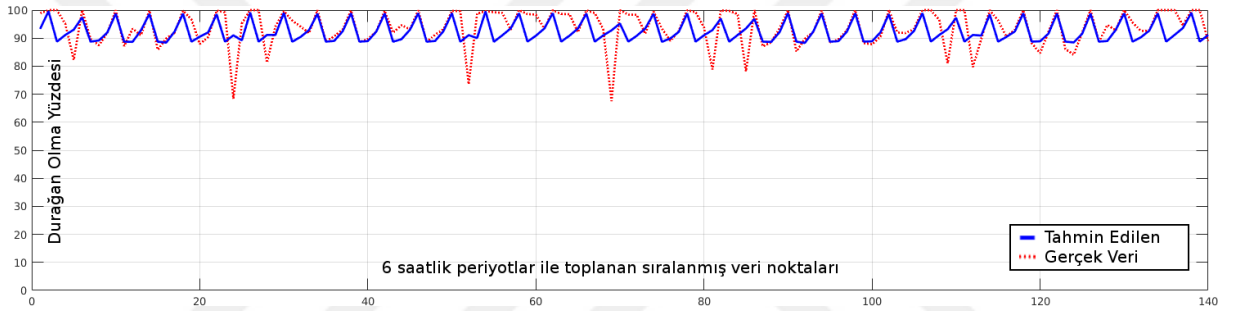


7 numaralı veri seti.

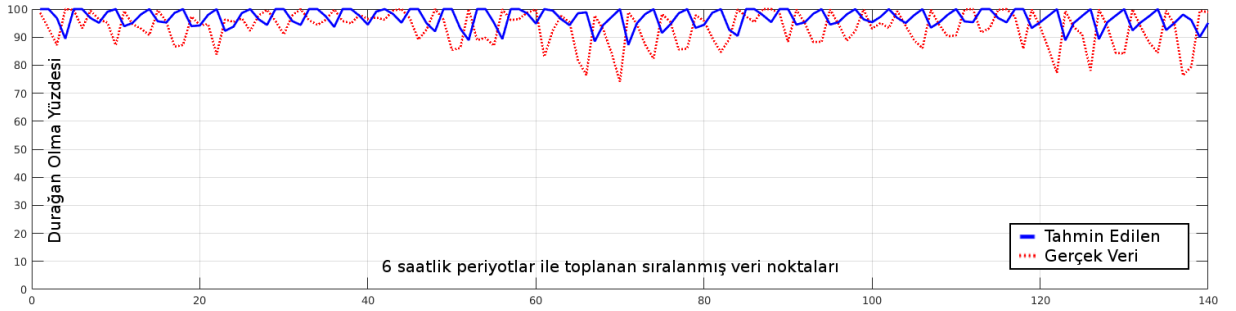
EK 1 (devam):



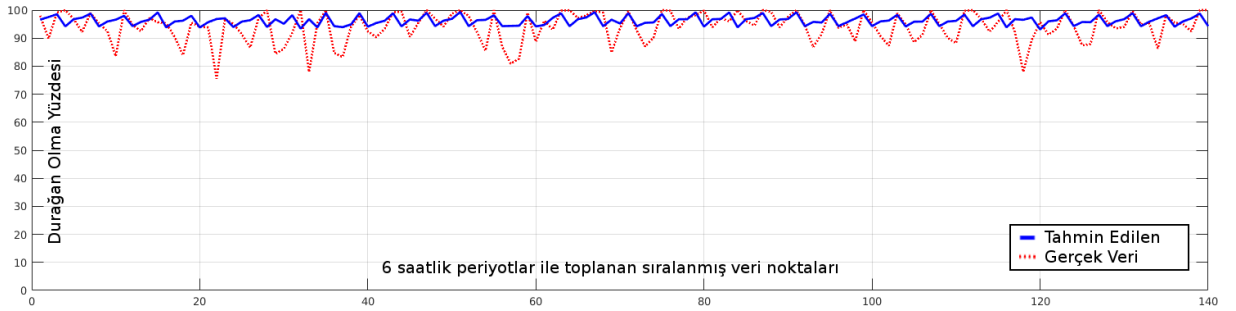
8 numaralı veri seti.



9 numaralı veri seti.



10 numaralı veri seti.



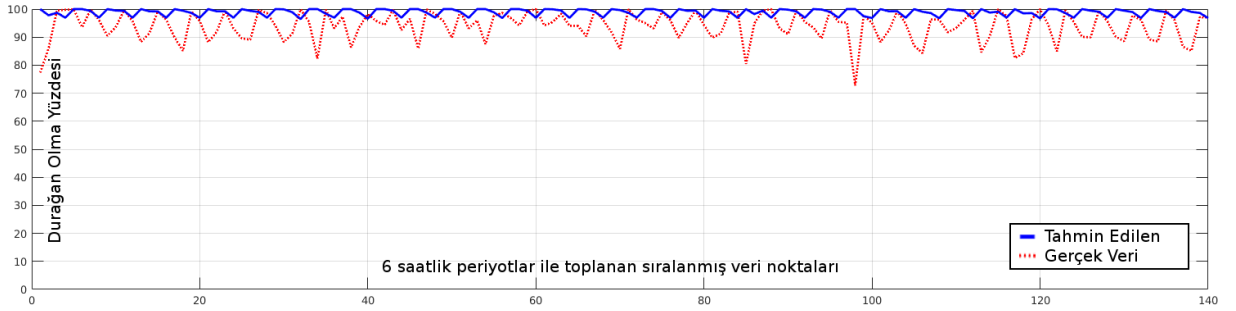
11 numaralı veri seti.

EK 1 (devam):

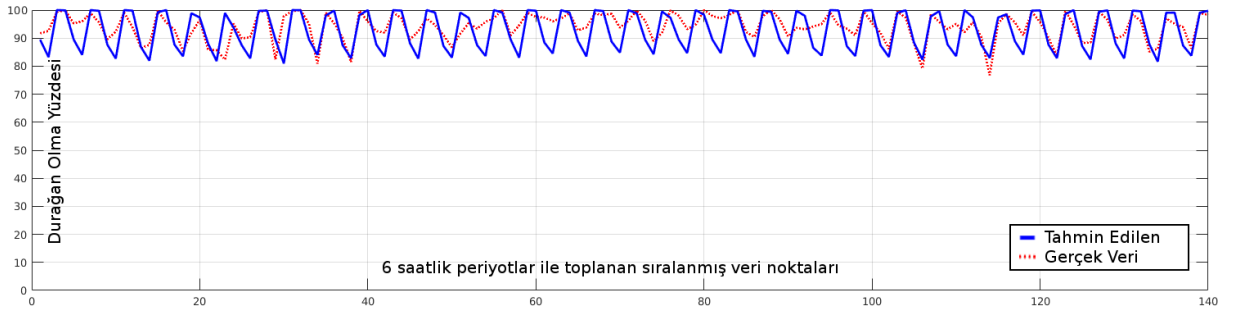
12 numaralı veri seti.



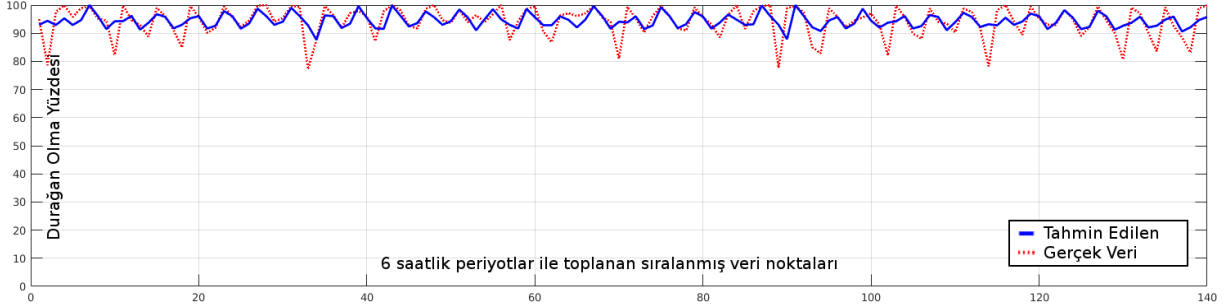
13 numaralı veri seti.



14 numaralı veri seti.



15 numaralı veri seti.

EK 1 (devam):

16 numaralı veri seti.



17 numaralı veri seti.



18 numaralı veri seti.

ÖZGEÇMİŞ

Kişisel Bilgiler	
Adı Soyadı	Gökhan ÖZOĞUR
Doğum Yeri	Fatih
Doğum Tarihi	02.01.1989
Uyruğu	<input checked="" type="checkbox"/> T.C. <input type="checkbox"/> Diğer:
Telefon	0535 698 41 43
E-Posta Adresi	gokhan@ozogur.net
Web Adresi	http://gokhan.ozogur.net



Eğitim Bilgileri	
Lisans	
Üniversite	Koç Üniversitesi
Fakülte	Mühendislik Fakültesi
Bölümü	Elektrik-Elektronik Mühendisliği Bölümü
Mezuniyet Yılı	2012

Yüksek Lisans	
Üniversite	İstanbul Üniversitesi
Enstitü Adı	Fen Bilimleri
Anabilim Dalı	Bilgisayar Mühendisliği Anabilim Dalı
Programı	Bilgisayar Mühendisliği Programı

Makale ve Bildiriler	
Bildiriler	
Ozogur, G., Erturk, M.A., ve Aydin, M.A., 2017, Prediction of Physical Activity Times Using Deep Learning Method, <i>International Telecommunications Conference</i> .	