
GAZİ ÜNİVERSİTESİ
Gazi
1926

ENDÜSTRİ 4.0 İÇİN BİR ANOMALİ TESPİT SİSTEMİ ÇERÇEVE GELİŞTİRİLMESİ

Cihan BAYRAKTAR

DOKTORA TEZİ
YÖNETİM BİLİŞİM SİSTEMLERİ ANA BİLİM DALI

GAZİ ÜNİVERSİTESİ
BİLİŞİM ENSTİTÜSÜ

AĞUSTOS 2022

ETİK BEYAN

Gazi Üniversitesi Bilişim Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmasında yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Cihan BAYRAKTAR
08 / 08 / 2022

ENDÜSTRİ 4.0 İÇİN BİR ANOMALİ TESPİT SİSTEMİ ÇERÇEVE GELİŞTİRİLMESİ

(Doktora Tezi)

Cihan BAYRAKTAR

GAZİ ÜNİVERSİTESİ

BİLİŞİM ENSTİTÜSÜ

Temmuz 2022

ÖZET

Dördüncü nesil endüstri devrimi olarak da ifade edilen endüstri 4.0, daha yüksek üretkenlik ve daha yüksek verimlilik elde etmek için gelişmiş üretim sistemlerinin ortaya çıkartılmasını ve kullanılmasını amaçlamaktadır. Endüstri 4.0 seviyesine ulaşılma gerekliliğinin sebeplerinden biri de daha hızlı ve hatasız üretim gerçekleştirilerek müşteri memnuniyetinin oluşturulması ile küresel pazarda önemli derece bir rekabet avantajının sağlanabilmesidir. Bu gelişmeler sayesinde, birbirlerine bağlanmış cihazlar vasıtası ile oluşturulan ağ yapıları şeklindeki üretim sistemleri, stabil çalışabilme konusunda, cihazların birbirleri arasında sağlıklı bir şekilde veri iletişimi gerçekleştirmelerine bel bağlamaktadır. Bu sebeple son teknolojiye uygun üretim sistemlerinin tasarlanması aşamasında güvenlik konusuna da azami derecede önem vermek gerekmektedir. Güvenlik konusunda alınabilecek önlemlerin en önemlilerinden biri, sistemin çalışması anında ortaya çıkabilecek cihaz ve veri hatalarının gerçek zamanlı olarak hızlı bir şekilde tespit edilebilmesi için anomali tespit yapılarının hazırlanması gerekliliğidir. Gelişmiş zeki üretim yapılarında anomali tespit sistemlerini kurabilmek için kullanılabilecek yöntemlerden biri makine öğrenmesi teknikleridir. Bu tez kapsamında, endüstri 4.0 için zeki bir üretim hattında makine öğrenmesi ile elde edilen öğrenme modellerini kullanarak, gerçek zamanlı anomali tespiti gerçekleştirebilecek ve tespitleri bir ekran vasıtası ile denetçiye bildirebilecek bir sistemin geliştirilmesi amaçlanmıştır. Bunun için de gerekli sentetik verinin elde edilebilmesi amacıyla bir üretim sistemi simülasyonu kurulmuştur. Simülasyon ile elde edilen statik ve dinamik verilerin değerlendirilmesi ile her cihaz için uygun öğrenme modelleri tespit edilmiş ve gerçek zamanlı anomali tespiti yapabilecek şekilde sistem ayarlanmıştır. Uygun öğrenme modellerinin tespit edilmesi için yapılan analiz çalışmaları sonucunda, tüm cihazlar için AutoML kütüphanelerinden olan Auto-Sklearn tarafından üretilmiş olan öğrenme modellerinin en iyi sonuca ulaştığı belirlenmiştir. Tezin hazırlanması aşamasında ulaşabildiğimiz kadarıyla, anomali tespiti konusunda yapılan çalışmaların çok büyük oranda, sadece verilerde anomali olup olmadığının tespit edilmesi üzerine olduğu, ancak tespit edilen anomali ile ilgili ayrıntılı bilgi verilmediği şeklinde olduğu görülmüştür. Tez kapsamında oluşturulmuş olan anomali tespit sistemimizde ise bunlardan farklı olarak, her cihaz için anomali tespitinin yapılması ile birlikte, hatanın hangi tür anomali olduğunun da belirlenmesi sağlanmıştır.

Bilim Kodu : 114607

Anahtar Kelimeler : Endüstri 4.0, zeki fabrikalar, anomali tespiti, makine öğrenmesi

Sayfa Adedi : 143

Danışman : Prof. Dr. Hadi GÖKÇEN

DEVELOPMENT OF AN ANOMALY DETECTION SYSTEM FRAMEWORK FOR INDUSTRY 4.0

(Ph. D. Thesis)

Cihan BAYRAKTAR

GAZİ UNIVERSITY

INFORMATICS INSTITUTE

July 2022

ABSTRACT

Industry 4.0, also referred to as the fourth-generation industrial revolution, aims to reveal and use advanced production systems to achieve higher productivity and higher efficiency. One of the reasons for the necessity to reach the industry 4.0 level is to achieve a significant competitive advantage in the global market by creating customer satisfaction by performing faster and error-free production. Thanks to these developments, production systems in the form of network structures created by means of devices connected to each other depend on healthy data communication between devices for stable operation. For this reason, it is necessary to give utmost importance to the issue of safety at the stage of designing production systems in accordance with the latest technology. One of the most important measures that can be taken regarding security is the necessity of preparing anomaly detection structures in order to quickly detect device and data errors that may arise during the operation of the system, in real time. One of the methods that can be used to establish anomaly detection systems in advanced intelligent production structures is machine learning techniques. Within the scope of this thesis, it has been aimed to develop a system that can perform real-time anomaly detection and report the detections to the auditor via a screen by using learning models obtained by machine learning in an intelligent production line for industry 4.0. For this, a production system simulation has been established in order to obtain the necessary synthetic data. By evaluating the static and dynamic data obtained through simulation, appropriate learning models have been determined for each device, and the system has been adjusted to perform real-time anomaly detection. As a result of the analysis studies carried out to determine the appropriate learning models, it has been determined that the learning models produced by Auto-Sklearn, which is one of the AutoML libraries for all devices, achieved the best results. As far as we can reach during the preparation of the thesis, it has been observed that the studies on anomaly detection are mostly focused on determining whether there is an anomaly in the data, but detailed information about the detected anomaly is not given. In our anomaly detection system, which was created within the scope of the thesis, unlike these, associated with anomaly detection is performed for each device, it is also possible to determine which type of anomaly the error is.

Science Code : 114607
Key Words : Industry 4.0, smart factories, anomaly detection, machine learning
Page Number : 143
Supervisor : Prof. Dr. Hadi GÖKÇEN

TEŞEKKÜR

Tez çalışmam boyunca ilgi ve katkılarıyla beni yönlendiren değerli danışmanım Prof. Dr. Hadi GÖKÇEN'e kıymetli desteğinden dolayı çok teşekkür ederim. Tez izleme komitemde yer alan kıymetli hocalarım Doç. Dr. Hakan ÇERÇİOĞLU ve Dr. Öğr. Üyesi Ziya KARAKAYA ile savunma jüri üyesi değerli hocalarım Doç. Dr. Recep Benzer ve Dr. Öğr. Üyesi Sami ACAR'a yönlendirmeleri ve değerli fikirlerinden dolayı teşekkürlerimi sunarım.

Doktora eğitimim süresince her zaman yanımda olduğunu hissettiğim değerli eşim Havva BULUT BAYRAKTAR'a gösterdiği sabırdan ve verdiği tüm destekten dolayı teşekkür ederim.

İÇİNDEKİLER

Sayfa

Sayfa

Sayfa

ÇİZELGELERİN LİSTESİ

Çizelge **Sayfa**

Sayfa

ŞEKİLLERİN LİSTESİ

Şekil **Sayfa**

Sayfa

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	**Açıklamalar**
ºC	Santigrat derece
sn	Saniye
W	Watt (Güç birimi)
V	Volt (Gerilim birimi)
A	Amper (Akım birimi)
Gr	Gram

Kısaltmalar	**Açıklamalar**
AutoML	Otomatik Makine Öğrenmesi (Automated Machine Learning)
BT	Bilişim Teknolojileri
DVM	Destek Vektör Makineleri
ERP	Kurumsal Kaynak Planlama (Enterprise Resources Planning)
FN	Yanlış Negatif Değerler (False Negative Values)
FP	Yanlış Pozitif Değerler (False Positive Values)
GBM	Gradient Boosing Machine
IoT	Nesnelerin İnterneti (Internet of Things)
KA	Karar Ağacı
k-NN	En Yakın Komşu (k-Nearest Neighbors)
KOBİ	Küçük ve Orta Boyutlu İşletme
LCD	Sıvı Kristal Ekran (Liquid Crystal Display)
LGBM	Light Gradient Boosing Machine
LR	Lojistik Regresyon
M2M	Makineler Arası İletişim (Machine to Machine)
MQTT	MQ Telemetri Aktarımı (MQ Telemetry Transport)
NAS	Sinir Mimari Araması (Neural Architecture Search)

NB	Naive Bayes
RFID	Radyo Frekanslı Tanımlama (Radio Frequency Identification)
RO	Rastgele Orman
SMAC	Sıralı Model Tabanlı Algoritma Düzenleyici (Sequential Model-Based Algorithm Configurator)
TN	Gerçek Negative Değerler (True Negative Values)
TP	Gerçek Pozitif Değerler (True Positive Values)
YSA	Yapay Sinir Ağları

1. GİRİŞ

Endüstrinin teknolojiyle bütünleşmesi ile devrimler ortaya çıkmıştır. Her bir devrim, endüstri uygulamalarına teknolojinin önemli değişiklikler yapacak şekilde entegre edilmesini temsil etmektedir. Gelinen son noktada, artık endüstrideki üretim sistemleri birbirlerine bağlantılı cihazlar arasında akan verilerin kontrolü, okunması ve işlenmesi ile kendi kendine işleyen otonom bir yapıya sahip olmaya başlamıştır. Teknolojinin her alanında olduğu gibi, bir ağ içerisinde transfer edilen verilerin güvenliklerinin sağlanması ve bu verilerin yorumlanarak anlamlandırılması da son derece önemlidir. Güvenlik kapsamında verilerin yorumlanması, özellikle sistem içerisindeki cihazların karralı çalışıp, çalışmadıklarını kontrol edebilmek için kullanılmaktadır. Bu kontrol sayesinde cihaz üzerinde veya genel yapı içerisinde oluşabilecek bir anomali, tespit edilebilmektedir. Anomalilerin hızlı tespiti de üretim sistemlerinde sorunun erkenden çözülmesine ve arıza maliyetlerinin en aza indirilebilmesine olanak sağlamaktadır.

Endüstri 4.0, özel kuruluşlar ve çeşitli üniversitelerin ortak çalışmaları sonucu, 2011 yılında Almanya'da ortaya çıkmış olan bir girişimdir. Dördüncü nesil endüstri devrimi olarak da ifade edilen girişim, endüstrinin daha üretken ve daha verimli çalışabilmesi için gelişmiş üretim sistemlerinin tasarlanmasını ve kullanılmasını amaçlamaktadır [1]. Endüstri 4.0 ile erişilmeye çalışılan gelişmiş üretim seviyesi sebebiyle, küresel pazarda da özellikle rekabet avantajının elde edilebilmesi için son derece önemli bir noktaya gelmiştir [2]. Endüstri 4.0 birçok teknolojik gelişmeyi bir çatı altında toplayarak, zeki üretim süreçleri ve gömülü sistemlerin birleştirilmesi ile yeni bir teknolojik gelişim ortaya çıkartmış ve bunu fabrikalardaki üretim sistemlerinin tamamına entegre edilmesine odaklanmıştır [3].

Endüstri 4.0'ın üç paradigma ile merkezi yönleri belirlenebilmektedir. Bunlar zeki ürün, zeki makine ve artırılmış operatör paradigmalarıdır. Zeki ürün, üretim modeline müdahale edebilen, kendi gerekli kaynaklarının talebini oluşturabilen bir durum haline gelmesidir. Zeki makine, makinelerin siber fiziksel sistemler haline dönüştürülmesi sürecini kapsamaktadır. Artırılmış operatör paradigması ise üretim sistemi çalışanlarının teknolojiye uyumunun sağlanarak, yeni yapıda daha verimli çalışabilmelerinin sağlanması olarak ifade edilmektedir [4].

Endüstri 4.0 yapısının ortaya çıkmasında, aynı çatı altında toplanmaya çalışılan kavramlar, zeki fabrikalar, siber fiziksel sistemler, nesnelerin interneti, makineler arası iletişim, yapay zeka, büyük veri, sanal gerçeklik, artırılmış gerçeklik, bulut bilişim, simülasyon teknolojileri ve siber güvenlik olarak sayılabilmektedir.

Yaşanan tüm bu teknolojik gelişmeler ile ortaya çıkan ağ yapıları, endüstri uygulamalarında sıklıkla kullanılmaya başlanmıştır. Ayrıca endüstri devrimleri vasıtası ile üretim sistemleri daha da gelişmiş ve otonom bir yapıya kavuşmaya başlamıştır. Otonom yapılar içerisinde cihazlar, düzgün çalışabilme konusunda birbirleri arasında gerçekleştirmeleri gereken veri iletişiminin sağlıklı olmasına bel bağlamaktadırlar. Eğer iki cihaz arasında aktarımı yapılan veride herhangi bir sebepten ötürü bir problem gerçekleşirse, tüm üretim hattının hatalı çalışması sorunu ortaya çıkabilir. Cihazlar arasında iletilen verilerde oluşabilecek olan hatalar, dışarıdan gerçekleşebilecek bir müdahale ile, kullanıcılar tarafından yapılabilecek hatalı veri girişi ile ve cihazların çalışma anlarında yaşanabilecek teknik arızalar sebebi ile gerçekleşebilmektedir. Bu sebeple, son teknolojik gelişmelere uygun bir üretim sistemi oluşturulmak istendiğinde, tasarım aşamasında güvenlik konusunun da tasarıma dahil edilmesi son derece önemlidir. Alınabilecek güvenlik önlemlerinin en önemlilerinden biri de sistemin çalışma anında ortaya çıkan hataların gerçek zamanlı olarak hızlı bir şekilde tespit edilmesidir. Bu kapsamda da anomali tespit sistemleri endüstri 4.0 için ön plana çıkmaktadır.

Anomali, bir cihazın normal düzenden önemli derecede farklı bir şekilde hareket edebilmesi olarak tanımlanmaktadır [5]. Sistemlerin normal çalışma süreçleri içerisinde meydana gelebilecek olan anormal uygulamalar sebebiyle ortaya çıkmaktadırlar [6]. Anomali tespiti de normal eylemler arasında ortaya çıkabilen ve veri yapıları içerisinde diğerlerine göre normalin önemli derecede dışında hareketler sergileyen kalıpların, veri ve veri örüntülerinin tespit edilme süreçleri olarak tanımlanmaktadır [7].

Zeki üretim sistemleri kapsamında anomali tespit sistemlerinin geliştirilmesinde en çok kullanılan yöntemlerden biri makine öğrenmesidir. Makine öğrenmesi de insanlar tarafından gerçekleştirilmesi çok zor olacak kadar büyük ve karmaşık yapıdaki faaliyetlerin uygulanabilmesine imkan tanımaktadır [8]. Makine öğrenmesi, hem normal hem de hatalı verileri içerisinde barındıran bir eğitim veri seti ile süreçte oluşabilecek hataları öğrenmeye

çalışan ve sonrasında sistemin normal çalışma düzeninde öğrenmiş olduğu hatalar ile karşılaştığında bunları tespit etmeye çalışan bir yazılım sistemidir [9]. Makine öğrenmesi ile anomali tespit sistemlerinin geliştirilmesinde dört farklı yaklaşım kullanılabilmektedir. Bunlar, mesafeye dayalı anomali tespiti, kümelemeye dayalı anomali tespiti, modele dayalı anomali tespiti ve sınıflandırmaya dayalı anomali tespiti olarak sayılabilir.

Bu tez kapsamında, endüstri 4.0 için zeki bir üretim hattında makine öğrenmesi ile elde edilen öğrenme modellerini kullanarak, gerçek zamanlı anomali tespiti gerçekleştirebilecek ve tespitleri bir ekran vasıtası ile denetçiye bildirebilecek bir sistemin geliştirilmesi amaçlanmıştır. Çalışma kapsamında, gerçek zamanlı anomali tespiti yapabilecek sistem çerçevesini geliştirebilmek ve testlerini yapabilmek amacıyla ayrık olay simülasyonuna dayalı bir patlamış mısır üretim sistemi simülasyonu ve anomali tespit sistemi oluşturduk. Sınıflandırma tabanlı anomali tespit yaklaşımı benimsendiği için, veri setleri eğitim ve test kümelerine bölünerek, makine öğrenmesi algoritmaları ve AutoML kütüphaneleri üzerinden analizlerin yapılamasını sağladık. İlk etapta statik veri ile yapılan analizler sonucunda, dinamik veri setleri ile analizlere devam ederek, veri setlerinin geçerliliklerinin doğrulanmasına yönelik çalışmalar yaptık. Statik veri setleri ile yapılan çalışmalarda başarı sağlayan öğrenme modelleri, dinamik veri setleri üzerinde de benzer performans göstererek geçerliliklerini doğrulamıştır.

Tezin hazırlanması aşamasında ulaşabildiğimiz kadarıyla, anomali tespiti konusunda yapılan çalışmaların çok büyük oranda, sadece verilerde anomali olup olmadığının tespit edilmesi üzerine olduğu, ancak tespit edilen anomali ile ilgili ayrıntılı bilgi verilmediği şeklinde olduğu görülmüştür. Tez kapsamında oluşturulmuş olan anomali tespit sistemimizde ise bunlardan farklı olarak, her cihaz için anomali tespitinin yapılması ile birlikte, hatanın hangi tür anomali olduğunun da belirlenmesi sağlanmıştır.

Çalışmanın devam eden ikinci bölümünde, Endüstri 4.0 kavramı, bileşenleri ve güvenlik riskleri ile ilgili literatür araştırmaları yapılmıştır. Üçüncü bölümde, anomali ve anomali tespiti kavramları ile ilgili ayrıntılı açıklamalar yapılmıştır. Dördüncü bölümde, makine öğrenmesi ile ilgili bilgi aktarımı yapılmış ve ardından tez kapsamında kullanılan makine öğrenmesi algoritmaları ve AutoML kütüphaneleri ile ilgili ayrıntılı bilgi verilmiştir. Beşinci bölümde, tez konusu ile ilgili gerçekleştirilmiş olan önceki çalışmalara atıflar

yapılmıştır. Altıncı bölümde ise çalışmamızın yöntemi, amacı, kapsamı ve sınırlılıkları ile ilgili bilgileri verilmiştir.

Bu tez çalışmasının uygulama bölümünde, çalışma kapsamında geliştirilmiş olan anomali tespit sistemi ayrıntıları ile açıklanmıştır. Bu bölümde, bir patlamış mısır üretim sistemi örneği olarak geliştirilmiş simülasyonun, ERP, Taşıma birimi, Depolama birimi, Dozlama birimi ve Üretim birimlerinin nasıl hazırlandığı ve üretim süreçleri içerisinde hangi görevlere sahip oldukları ile ilgili ayrıntılı bilgilere yer verilmiştir. Sonrasında ise simülasyon tarafından üretilmiş olan sentetik veriler üzerinde analizler gerçekleştirilerek gerçek zamanlı bir anomali tespit sisteminin nasıl çalıştığına dair açıklamalar verilmiştir.

2. ENDÜSTRİ 4.0 VE TEMEL KAVRAMLAR

Endüstriyel süreç gelişimi, ilk olarak 18. yy. da buhar makinelerinin bulunması ile başlayan Endüstri 1.0 devrimi olarak gösterilmektedir. İlerleyen süreçte 20. yy. ın başlarında elektrik enerjisinden faydalanılarak seri üretime geçilmesi Endüstri 2.0 devrimini getirmiştir. Bunu, elektronik ve bilgisayar teknolojisinin gelişmesi sonucunda üretimin dijitalleşmesi devrimi olan Endüstri 3.0 izlemiştir [10]. Gerçekleşmekte olan Endüstri 4.0 devriminde ise, üretimde bilgisayarlaşma süreçlerinin çok daha yoğun bir şekilde yaşanması beklenmektedir. Yüksek teknolojinin üretime adapte edilmesi ile birlikte daha yoğun bilgisayar, daha yoğun yazılım, daha yoğun beyin fırtınası ile karar verme süreçleri ve zeki sistemlerin işletme süreçlerine kullanılmaları beklenmektedir [11].

Endüstri 4.0 aslında, üretimde yüksek seviyede esneklik, gerçek zamanlı izlenebilen yüksek verimlilik oranları ve düşük fire miktarı elde etmek için endüstriyel internet, geleceğin fabrikaları, nesnelerin interneti, fiziksel internet, hizmetlerin interneti ve siber fiziksel sistemleri gibi girişimleri içeren bilgi ve iletişim teknolojileri üzerine tasarlanmış, Alman Hükümetinin ileri teknoloji stratejileri içerisinde tanımlanmış bir gelecek vizyonudur. Siber bağlantılı üretim sistemleri, süreçlerin optimize edilmesini ve verimliliğin artmasını sağlar, ayrıca üretici ve sanayi şirketlerinin kendi iş süreçleri ile ilgili yolları da değiştirme potansiyeline sahiptir [12].

Endüstri 4.0, 2011 yılında üniversiteler ve özel şirketler ile birlikte Alman Federal Hükümeti tarafından bir girişim olarak icat edilmiştir. Stratejik bir program olarak ortaya çıkan bu programın amacı, endüstrinin üretkenliği, etkinliği ve verimliliğini artırmak ve gelişmiş üretim sistemleri ortaya çıkartmaktır. Bu yapı, ürün yaşam döngüsüne katkı sağladığı bilinen bir dizi teknolojinin bir çatı altında bir araya getirilerek, ortak bir yapı içerisine entegre edilmesini kapsamaktadır. Endüstri 4.0, gelişmiş üretim veya zeki üretim yapısında, esnek hatların oluşmasına imkan tanıyan ve bu sayede, çok çeşitli ürün türü ve değişen şartlar doğrultusunda üretim süreçlerinin otomatik ayarlandığı bir sistem olarak kullanılmaktadır [1].

Endüstri 4.0, teknolojileri ile daha hızlı ve daha yüksek verim elde edilebilen üretim süreçleri gerçekleştirildiğinden dolayı, küresel pazarda büyük bir öneme sahip olmuştur.

Endüstri devrimlerinin en başından itibaren gerçekleşen süreçler sonucunda büyük oranda teknik gelişmeler ortaya çıkmıştır. Genel itibari ile Endüstri 4.0, bir çatı altında çalışan uygulamalardan oluşan bir birleşim olarak ifade edilmektedir. Bunların bir kısmı; büyük veri, otonom robotik, nesnelerin endüstriyel interneti, simülasyon, yapay zeka, siber güvenlik, siber fiziksel sistemler, bulut sistemleri ve zeki fabrikalar olarak listelenebilmektedir [2].

Endüstri 4.0 terim olarak, tüm endüstriyel sistemde kullanılan cihazların birbirleri ile bağlantılı bir yapı içerisinde bulunmalarını temsil etmektedir. Endüstri 4.0 devrimi, zeki üretim süreçleri ile gömülü sistemlerin birleştirilmesi ile yeni bir teknolojiyi elde etmek ve devamında üretim süreçleri gibi işletme içerisindeki tüm kademelere uygulanmasını amaçlamaktadır. Günümüz işletmeleri, birçok gelişmiş ürünün üretilmesi aşamasında büyük veri yığınlarının cihazlar arasında aktarılması ve işlenmesi konusunda sorunlar yaşamaktadırlar. Yeni nesil endüstri devrimi teknolojilerinden gösterilen siber fiziksel sistemler, yaşanan bu sorunların çözümü için zeki birimleri iç içe girdiği yapılardan oluşan yapay zeka sistemleri ve büyük veri yığınlarının güvenle depolanabileceği bulut bilişim olanakları gibi imkanlar sunabilmektedir [3].

Dördüncü endüstri devrimi kapsamında değerlendirilen birçok uygulamanın temelinde, bilgisayar destekli merkezi yönetim, veri toplama ve işleme mekanizmaları bulunmaktadır. Bu mekanizmalar ile kurulan sistem, her geçen gün siber suçlara karşı ilgi odağı olmaya devam etmektedir. Bundan dolayı işletmeler, kendi iç ve dış süreçlerinin aksamasını engellemek, iş verimliliğini kaybetmemek ve sistem içerisinde üretilen ve işlenen verilerin yetkisiz kişilerin eline geçmesini engellemek için çeşitli bilgi güvenliği uygulamalarına ihtiyaç duymaktadırlar [13].

Endüstri 4.0 devriminde, zeki fabrikalar tarafından temsil edilen fiziksel sistemler ve bilişim teknolojilerinin etkileşimi, birbirine bağlı olan tüm sistemler arasında çok büyük miktarlarda gerçek zamanlı veri alışverişine imkan sağlamıştır. Bu sayede, üretim alanında anomali tespit sistemleri için gereklilik ortaya çıkmıştır. Anomali tespit sistemlerinde, üretim sistemi içerisinde çalışan cihazlardan toplanan veriler, bilişim sistemleri içerisine dahil edilmektedir. Sisteme aktarılan veriler üzerinden anlık değerlendirme yapılarak, herhangi bir anomali tespit edildiği takdirde, sistemde oluşacak düşük performans ve yüksek hata ihtimallerini hızlı bir şekilde engellemek için operatörlere gerekli bilgilerin

aktarım işlemleri sağlanabilmektedir. Bu sayede yüksek oranda kesintisiz ve performanslı bir üretim sisteminin işleyişi mümkün olabilmektedir [14].

Daha önce gerçekleşmiş olan üç endüstriyel devrim, yaklaşık olarak 200 yıl sürmüştür. İlk olarak buhar motorlu mekanik tezgahlar, üretimde önemli bir değişimi başlattı. Ortaya çıkan bu değişimin sonucunda kumaş üretiminin merkezi fabrikalar lehine, ev ortamlarını terk etmiş ve sonucunda verimliliğin ciddi oranda artmış olması örnek olarak gösterilebilir. İkinci endüstri devrimi, yaklaşık 100 yıl sonra Ohio'daki kesim tesislerinde başlamış ve Amerika Birleşik Devletleri'nde Ford markasına ait T model otomobillerin üretimi ile birlikte zirveye ulaşmıştır. Kesintisiz üretim hatlarının gelişimi hem iş bölümünün oluşturulması hem de konveyör hatlarında gerçekleşen verimlilik patlaması ile gerçekleşmiştir. Üçüncü endüstri devrimi ise 1969 yılında Modicon firması tarafından duyurulan otomasyon sistemlerinin programlanabilmesine olanak sağlayan ilk programlanabilir mantık denetleyicisi ile başlamıştır. Bu programlanabilir örnek hala günümüz otomasyon sistem mühendisliğini yönetmekte ve bir hayli esnek ve verimli otomasyon sistemlerine öncülük etmektedir [15].

Dördüncü endüstri devriminin, [16] tarafından kaleme alınmış olan makale ile başladığı belirtilmektedir. Bu makaleye göre, Endüstri 4.0 otomasyondaki gelişmeler ve bunların yanında zeki gözlem ve karar verme aşamalarını da kapsadığı ifade edilmektedir [17].

Bütünleşmiş sanayi adı ile bilinen Endüstri 4.0 son zamanlarda herkes tarafından dile getirilmektedir. Büyük işletmeler, orta sınıf işletmeler ve kamunun belli kesimleri Endüstri 4.0 ile ilişkili yeni fırsatları incelemeye devam etmektedirler. Dördüncü endüstri devrimine karşı oluşan bu ilgi, sadece en önemli uluslararası bir fuar olan Hannover Fuarının odak alınmasıyla değil, ayrıca Alman hükümetine bağlı birkaç bakanlık tarafından destek sağlanan BT Zirvesindeki profilini de önemli ölçüde yükseltmiştir. Fakat sanayiyi kuşatan bu büyük yayılımda, büyük veri, bulut bilişim, siber fiziksel sistemler, RFID mimarileri, nesnelerin ve hizmetlerin interneti ve makinalar arası iletişim gibi Endüstri 4.0 terimleri birçok pazarlama stratejisinin niyeti gibi özensiz bir şekilde kalmıştır. Bu düzensiz Endüstri 4.0 tanımı defalarca, sonrasında karşılaşılmayan ve hayal kırıklığına neden olan fazla abartılmış beklentilere yol açmıştır. Tedarikçileri, Endüstri 4.0 kapsamında değerlendirilen teknolojilerin, makine otomasyonunu tamamlayıcı nitelikte olduklarını ve

bu sayede üretim maliyetlerinin tüketicilerin lehine daha uygun seviyelere ulaştırılabileceğini ifade etmişlerdir [18].

Endüstri 4.0'a ait merkezi yönler, üç paradigma ile daha net bir şekilde belirlenebilir. Bunlar; zeki ürün, zeki makine ve artırılmış operatördür. Zeki ürünün ana fikri, sistemin aktif bir bölümüne iş parçasının rolünü yaymak olarak ifade edilmektedir. Bu paradigmada ürünlerin, operasyonel verilerin ve gereksinimlerin doğrudan bir bireysel inşa planı olarak saklandığı hafızanın yerini aldığı belirtilmektedir. Yani bu durumda ürün, kendisi için gerekli kaynakları talep eden ve kendi üretim süreçlerini yönlendiren bir durum haline gelmektedir. Bu, modüler üretim sistemlerinde, ürünlerin kendi kendilerini konfigüre edebilme seviyesine ulaşabilmeleri için sahip olmaları gereken ön şarttır. İkinci paradigma olan zeki makineler, makinelerin siber fiziksel sistem olma süreçleri olarak tanımlanmaktadır. Bu süreçte geleneksel üretim hiyerarşisi, siber fiziksel sistemler tarafından etkinleştirilen kendi kendine örgütlenme sistemine dönüştürülecektir. Ayrıca üretim hatları, oldukça esnek seri üretim koşulları altında, en küçük parti büyüklüğünü bile üretebilecek kadar esnek ve modüler bir seviyeye ulaşacaktır. Üçüncü paradigma olan artırılmış operatör, modüler üretim sistemlerinin mücadeleci çevresi içerisinde çalışanlar için teknolojik destek sağlamayı hedef almaktadır. Endüstri 4.0 çalışanın olmadığı bir üretim tesisi oluşturmayı amaçlamamaktadır. Çalışanlar, giderek daha da zorlu bir ortam haline gelen iş çevresinin, azami ölçüde uyum sağlayabilen en esnek parçaları olarak kabul edilmektedirler [4].

Dördüncü endüstri devrimi, yeni nesil otomasyon sistemlerini, veri akış kontrollerini ve üretim alanında geliştirilen teknolojileri bir arada bulunduran bir kavramdır. Sahip olduğu ayırt edici unsurlar şu şekilde ifade edilebilmektedir [19]:

- Hız**:** Özellikle endüstriyel alanda kullanılan teknoloji çok hızlı bir gelişim göstermektedir. Hemen hemen her gün yeni bir gelişime tanıklık edilebilmekte ve her yeni gelişim, bir sonraki teknolojik ilerlemenin temelini oluşturmaktadır.
- Genişlik ve Derinlik**:** Günümüz iş ortamında gerçekleşen gelişimler, dijital dünya üzerinde oluşmaktadır. Ancak tüm bu gelişmeler, sadece üretim ortamını etkilemekle kalmamakta, iş dünyası, toplum ve bireyin günlük yaşamında dahi köklü değişikliklere sebep olabilmektedir.

- Sistem Etkisi**:** Bu yeni süreç, şirketlerin, faaliyetlerin ve hatta ülkelerin yapısını değiştirmektedir. Kullanılan bütün sistemler bütünsel olarak değişime zorlanmaktadır.

Endüstri 4.0'ın diğer endüstri devrimleri içerisinde planlı olarak gerçekleşen ilk devrim olacağı belirtilmektedir. Yeni nesil endüstri, dijitalleşen üretim sistemlerinin çıktıları ve bileşenleri tarafından şekillendirilecektir. Bu durum sayesinde, üretim sistemlerinde kullanılan tüm fiziksel bileşenler, makineler arası iletişim sistemine entegre edilecektir. Özellikle üretim sistemlerinde uygulanacak olan dijitalleşme süreci, optimum çalışma, kişiselleştirilmiş ürünler ve esnek üretim yapılarının ortaya çıkmasını sağlayacaktır [20].

Endüstri 4.0 devrimi sonrasında, zeki makinelerin üretim sistemlerindeki kontrolü sağladıkları bir dönem başlamış olacaktır. Bu da dönem içerisinde istihdam kapasitesinin önemli ölçüde azalacağını göstermektedir. Çalışanların refah payları üzerinde iyileştirme çalışmaları yapılarak bu sorun için çözüm üretme yoluna gidilebilir. Ancak bu konu hakkında çalışanlar için hem psikolojik hem de sosyolojik açıdan gerekli değerlendirmeler yapılarak açıklığa kavuşturulmalıdır [21].

2.1. Endüstri 4.0 ile İlişkili Kavramlar

Endüstri 4.0 sürecinin başlaması ile yüksek ekipmana sahip makine ve robotların üretim sistemlerine dahil olması ve en ufak gelişmeye uyum sağlamaları, üretimde verimliliği artması ve maliyetin azalması, otomasyon sistemlerinin devreye girmesi ile çalışan sayısında azalma olması, ancak üretim sistemlerinde görev yapan çalışanların eğitim seviyelerinde artış olması, veri işlemenin zeki sistemler sayesinde günümüz şartlarına göre çok daha kolay bir hale gelmesi, müşteri memnuniyetinin en üst seviyeye ulaşması ve nesnelerin interneti ile yeni iş modellerinin geliştirilmesine imkan verilmesi gibi özelliklerin gelişmesi beklenmektedir [11].

2.1.1. Zeki fabrikalar

Endüstri 4.0'ın yapı taşlarından biri olan zeki fabrikalar kavramı, akademisyenler ve uygulayıcılar tarafından farklı açılardan tanımlanmaktadır. Ancak yine de bu konuda yapılmış tutarlı ve paylaşılan bir tanım bulunmamaktadır. Yapılan tanımların birçoğuna

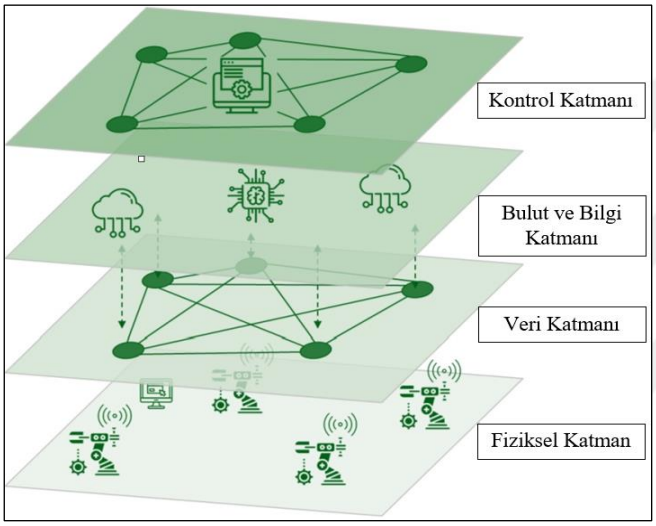
göre zeki fabrikalar, zeki üretim sitemlerinin oluşturulabilmesi için tüm iletişim sistemlerini, bilişim alt yapısını ve denetim süreçlerini içerisinde barındıran ve siber fiziksel sistemlerin merkezine konumlandırılmış bir kavramdır. Zeki fabrika sistemleri, özerk ve zeki olarak nitelendirilen makinelerin ve ürünlerin son derece esnek davranabilmesi ile oluşan üretim sistemi ve kendi kendini uyarlayabilen üretim süreçleri ile fiziksel varlıkları elde edebilmek ve bilişim sistemlerinin dikey etkileşimini sağlayabilmek için daha yüksek seviyeli üretime dayalı siber fiziksel sistemler oluşturmayı amaçlamaktadır [22].

Üretimin tamamen sensörler ve özerk sistemler ile donatılmış uygulamalar ile gerçekleştirilmesi olarak tanımlanmaktadır. Üretilen ürünlerin ve fabrikaların, bütünsel olarak dijitalleştirilmiş olan modelleri ve her yerde programlamanın çeşitli teknolojilerinin uygulaması ile ilişkili olan zeki teknoloji kullanılarak geliştirilmiş olan, kendi kendini idare edebilen zeki fabrikaların geliştirildiği söylenmektedir [23].

Gelecekte kullanılması planlanan zeki fabrika sistemlerinde önceki dönemlere göre daha az miktarda fire ile müşteri taleplerine anlık olarak cevap verilebilmesi hedeflenmektedir. Günümüz tüketim yapısı eskiye göre sadece basit farklılaşma geçirmekle kalmayıp sürekli olarak değişen bir niteliğe sahip olmuştur. Geleceğin zeki fabrikaları ürün çeşitliliğini yüksek seviyelere ulaştırmanın yanında, değişen müşteri taleplerine göre anında üretim imkanlarını da getirebilecektir [17].

Zeki fabrikalar, sensörler ile donatılmış, veri üretebilen, aktarımını yapabilen, işleyebilen ve saklayabilen makinelere sahip bir yapıdır. Bu makineler, daha önceden belirlenerek sisteme işlenmiş olan süreçleri gerçekleştirmek için, oluşturulmuş olan sensör ağı aracılığı ile birbirleri arasında iletişim halinde olurlar. Makineler, aynı amaç doğrultusunda organize bir yapı ortaya koyarlar ve gerekli durumlarda kendilerini yapılandırabilirler. Sistem bu sayede bir ana yazılım ve bir insan ikilisi tarafından sürekli olarak takip altında tutulur ve yazılımdan gelen yönergeler doğrultusunda işlemlerine devam eder. Zeki fabrika yapıları sayesinde, üretim sistemleri artık bir faaliyet düzlemi şeklinde görülmek yerine, dört ayrı katmana sahip bir yapı olarak değerlendirilmektedir. Bu katmanlar; fiziksel katman, veri katmanı, bulut ve bilgi katmanı ve kontrol katmanı olarak nitelendirilmektedir (Şekil 2.1). Zeki fabrika yapısı içerisinde kullanılan bütün makinelerin bir çatı altında, belirli bir düzene göre konumlandırılması fiziksel katman olarak değerlendirilmektedir. Veri

katmanında makine ve bulut arasında çift yönlü veri aktarımı gerçekleştirilmektedir. Aynı zamanda hangi verilerin gönderilip alınacağı ve hangi hızlarda gönderim işlemlerinin yapılacağı konuları da bu katmanda gerçekleştirilmektedir. Sistem içerisinde üretilen ve taşınan veriler, bir süreliğine de olsa bulut ve bilgi katmanında depolanır. Ayrıca burada depolandıkları süre içerisinde çeşitli analitik süreçlerden geçirilebilirler. En üst katman olan kontrol katmanında ise yapının denetimini gerçekleştiren ana yazılım bulunmaktadır. Bu kısımda gerekli görüldüğünde ana yazılım üzerinden sisteme insan müdahalesi uygulanabilir [24].



Şekil 2.1. Dört katmanlı zeki fabrika yapısı [24]

Bazı çalışmalarda, zeki fabrikaların oluşturulabilmesi için bilgi teknolojilerinin üretim sistemlerine entegrasyonun sağlanması ve kullanım ağının genişlemesi gerekliliğine olan önem vurgulanmaya çalışılmıştır. Genel anlamda, bunun gerçekleşebilmesi için, zeki bir üretim sisteminin sahip olması gereken 3 temel unsurdan bahsedilmektedir. Bunlar; şeffaflık, otomasyon ve sürdürülebilirliktir. Zeki fabrikaların, sistem yapısı içerisinde birlikte çalışabilme özelliğine sahip olması gerekmektedir. Sistem içerisinde akan verilerin toplanması, üretim süreçleri için gerekli düzenlemelerin yapılması ve tüm hizmetlere erişim süreçleri zeki fabrika mimarileri içerisinde değerlendirilmesi gereken koşullar olarak belirtilmektedir [25].

Büyük veri, nesnelerin interneti (IoT), siber fiziksel sistemler gibi yenilikçi kavramların yükselişleri ve çeşitli teknolojilerin geliştirilmesi yeni dönemde endüstri yapısını üst seviyelere taşımıştır. Bunların sonucunda zeki üretim sistemlerinin oluşturulmasını hedefleyen makineler arası iletişim (M2M) yapısı ile zeki fabrikalar ortaya çıkmıştır. M2M

sayesinde zeki fabrikalar, bilgilerin bağımsız olmadığı grup etkileşimlerini gerçekleştirmektedirler. Birimler arasında bilgilerin kaynaşması ve çarpışması aracılığı ile zeki üretim süreçlerinin geliştirilmesi sağlanabilmektedir [26]. Zeki fabrikalar, ürünlerin kısa yaşam döngülerinden, deneyimli çalışan eksikliğinden, ülkelerce yürütülen çeşitli çevre düzenlemelerinden ve sürekli olarak artan müşteri taleplerinden kaynaklanan sorunların, hızlı ve hatasız bir şekilde çözüme kavuşturulabilmesi için geliştirilmektedir [27].

Zeki fabrikalar, gün geçtikçe daha da karmaşık hale gelen bir dünyada, dinamik ve hızlı değişen şartlara sahip bir üretim sistemi için, ortaya çıkabilecek sorunları çözebilecek esnek bir üretim sistemi çözümüdür. Bu çözüm aynı zamanda, gereksiz iş gücü ve kaynak israfının önüne geçebilmek amacıyla kullanılması gereken yazılım ve donanım birimlerinin kombinasyonunu da ilgilendirmektedir. Ayrıca zeki sistemlerin gerçekleştirmesi gereken görevlerden biri olan endüstriyel ortam ve çevre arasındaki bağlantının da sağlıklı bir şekilde kurulması ve yönetilmesi konuları ile de ilgilenmektedir [28].

2.1.2. Siber fiziksel sistemler

Siber fiziksel sistemler, fiziksel dünya ile siber dünyanın internet alt yapısı aracılığı ile birbirine bağlanması olarak tanımlanmaktadır. Bu sistemler sensörler ile donatılmışlardır ve bu sayede fiziksel dünyada gerçekleşen hareketler internet hizmetleri ile toplanmaktadır. Bunu sonucunda elde edilen veriler nesnelerin birbirleri ile etkileşimlerinde kullanılmaktadır. Sürekli değişken halde bulunan veriler, siber fiziksel sistemler vasıtası ile eşzamanlı olarak bir bulut sisteminde birbirlerine bağlanmaktadırlar. Sosyoteknik sistem içerisinde bulunan siber fiziksel sistemler, üretim süreçlerinde insanımsı makine ara yüzlerini kullanmaktadırlar [17].

Fiziksel ve dijital düzeyin birleşimi olarak ifade edilebilir. Bu birleşme, ürünler ile birlikte üretim seviyesini de kapsıyorsa, fiziksel ve dijital aşamaların artık kolay bir şekilde ayırt edilemediği sistemlerin ortaya çıktığı söylenebilir. Mekanik araç ve gereçlerde fiziksel olarak gerçekleşen aşınma ve yıpranmaların altında yatan sebepler olan stres, üretim süresi gibi süreç parametreleri dijital olarak kaydediliyor olması işletmelerin bakım birimleri açısından bir örnek olabilir. Bundan dolayı, sistemin gerçek durumu, fiziksel nesne ve dijital işlem parametrelerinin sonucu olarak ele alınabilir [23].

Siber fiziksel sistemler, endüstriyel anlamda bilgi ve iletişim teknolojilerinin gelişimi içerisinde sahip olduğu konum sebebiyle, çeşitli çalışmalarda geleceğin fabrikaları için önemli bir yapı taşı olarak görülmektedir. Siber fiziksel sistemler özellikleri gereğince, kendisini çevreleyen fiziksel yapı ile bağlantı içerisinde olan, internet alt yapısı sayesinde veri erişim ve veri işleme imkanları sağlayan ve bu imkanları kullanan, işletme süreçlerinde iş birliği yapılmasını destekleyen bir yapıdır. Siber fiziksel sistemler içerisinde kullanılan sensörler, makineler ve robotlar gibi ekipmanlar, sistem içerisinde zeki bir topluluğun ortaya çıkmasını sağlamaktadır. Elde edilen bu yetenek sayesinde siber fiziksel sistemler, elde edilen ham verileri eyleme geçirerek süreçlere aktarabilir, kullanıcıların gerçekleşen işlemler hakkında bilgi sahibi olmalarını ve anlamalarını kolaylaştırır ve kanıtlara dayalı karar verme yapısına destek verdiğinden dolayı üretim sistemine esneklik katar [29].

Alanda yapılmış olan bazı çalışmalarda, zeki üretim sistemlerinin yeterince kabullenilmesi ve kullanılabilmesi için siber fiziksel sistemlerin önemini vurgulamaktadırlar. Siber fiziksel sistemler, hesaplama, iletişim ve denetim fonksiyonlarının birleştirilmesi ile yaygın bir şekilde dağıtılmış olan gömülü sistemlerde geri bildirim yapısının etkin kullanımını sağlayan çok disiplinli bir teknoloji olarak ifade edilmektedir. Endüstri 4.0 yapısının, kilit teknolojilerinden biri olan siber fiziksel sistemler, nesnelerin interneti, M2M, büyük veri ve bulut bilişim teknolojilerinin daha etkin ve hızlı bir şekilde kullanılmasına imkan vermektedir [25].

2.1.3. Nesnelerin interneti (IoT)

IoT tanımına geçmeden önce, genel olarak yapıyı oluşturan etkin bileşenlerin neler olduğunu listelemek yararlı olacaktır. Temel anlamda IoT, aşağıda belirtilmiş olan araçlar tarafından kuvvetlendirilmiş fiziksel nesneler ağı olarak belirtilebilir [30]:

- Sensörler**:** Gerekli verileri toplamak için,
- Tanımlayıcılar**:** Verilerin hangi kaynaklardan elde edileceğinin tanımlanması için,
- Yazılım**:** Toplanan veriler üzerinde gerekli analiz işlemlerinin yapılabilmesi için,

- İnternet Bağlantısı**:** Cihazlar arasında iletişim hattını kurmak ve bilgilendirmek için kullanılmaktadır.

IoT terimi içerisinde internet kavramı, ya web'in bir metaforu olarak nesnelerin birbirlerine bağlanmasında, hizmetlerin kullanılmasında ve veri üretilmesinde ya da daha katı bir tanımla, bir IP protokolü yığını olarak zeki nesneler tarafından kullanılmaktadır. IoT, nesne – nesne ve insan – çevre arasındaki bağlantıları tanımlamaktadır. Bunun dışında Her Şeyin İnterneti (IoE), İnsanların İnterneti (IoP) ve Verilerin İnterneti (IoD) kavramları da IoT çatısı altında değerlendirilmektedir. Bu terimler aracılığıyla, IoT'nin teknolojik araçlar ve teknolojik olmayan nesnelerden oluştuğu ortaya çıkmaktadır [31].

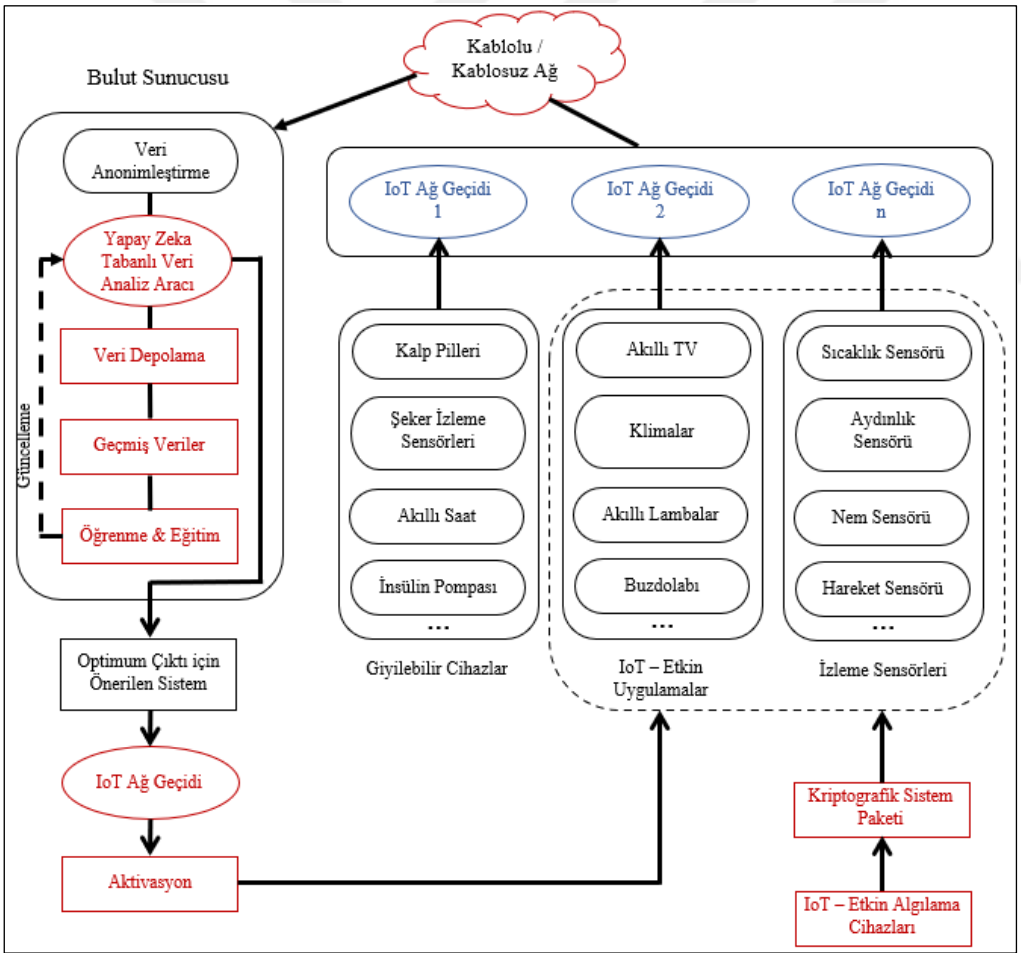
IoT, fiziksel nesnelerin internet ağına tam olarak entegre edildiği ve bilgi ağını kullanarak iş süreçlerinde aktif rol aldıkları bir ağ yapısı olarak tanımlanabilmektedir. IoT, bu anlamda mal ve hizmet alışverişlerinin küresel pazarda daha kolay ve hızlı bir şekilde gerçekleştirilmesini sağlayan bir mimari olarak anılmaktadır. İşlev olarak, fiziksel nesneler ile bilgi teknolojileri arasında bulunan boşluğu doldurmaya ve bu sayede daha şeffaf küresel iş ağları oluşturarak verimliliği artırmaya yönelik hizmet etmektedir [32].

IoT ile ilgili çözümler, günümüzde neredeyse tüm alanlara yayılmış durumda olduğu için uygulama alanları çok çeşitlenmiştir. En çok öne çıkan uygulama alanları olarak, zeki ve bağlantılı üretim sistemlerinin geliştirilmesi süreçlerinin genel anlamda endüstri 4.0 kapsamına alındığı zeki endüstri yapısıdır. Zeki ev ve binalar konusunda, zeki termostatlar ve gelişmiş güvenlik sistemleri ön plana çıkarken, zeki enerji uygulamaları için de odak noktaları oluşmaya başlamıştır. Bunların dışında araç takip sistemleri, mobil bilet sistemleri, hasta takip sistemleri gibi alanlarda da kolay ve verimli yönetim için IoT'den önemli derecede söz edilmektedir [33].

Bilgi teknolojileri açısından konuya bakıldığında, nesneler olarak ifade edilen her şeyin arasındaki bulunan dijital bağlantıları gerçekleştirmek ve bu bağlantıları güvenilir ve kalıcı bir şekilde sürdürmekle ilgili birçok şey bulunmaktadır. Cihazlar arasında verilerin gerektiği şekilde aktarılması, donanım arızalarının zamanında tespit edilebilmesi, yazılım hatalarının giderilebilmesi, veri bütünlüğünün korunabilmesi ve veri paylaşımının güvenilir bir şekilde yönetilebilmesi son derece önemli konulardır. Tüm bu karmaşık yapı, hemen hemen sonsuz sayıda bulunan üretici ve ürün kombinasyonlarından oluşan farklı

bilgi teknolojileri sistemleri ve IoT çerçeveleri ile sağlanmalıdır. Tabi ki bu sistem içerisinde güncelleştirme yamalarının ve sistem yükseltmelerinin, kesintiye sebep olmadan gerçekleştirme zorluğu da bulunmaktadır [34].

Şekil 2.2'de, IoT veri akış diyagramı verilmiştir. Diyagrama göre veriler, öncelikler IoT özellikli cihazlardan ve IoT cihazlarından gelmektedirler. Ardından ağ geçidi aracılığı ile bulut sistemine aktırılırlar. Bulut sistemi içerisinde bulunan sunucuda, gelen veriler üzerinde eğitim ve öğrenme süreçleri çalıştırılmaktadır. Amaç, sistem kapsamında gerçekleştirilecek optimal eylemler için en iyi modellerin veya sistemlerin önerilmesini sağlamaktır. Öneriler sonucunda, aktuatorler aracılığı ile IoT cihazları harekete geçirilmektedir [35].



Şekil 2.2. IoT akış diyagramı [35]

Aralarında sürekli olarak iletişim halinde bulunan cihazlar, alanın daha etkin değerlendirilmesine katkı sağladığından dolayı, daha verimli bir yerleşim düzeni oluşturulabilmektedir. Nesnelerin interneti teknolojisi sayesinde, cihazlar birbirleri ile

aralarında parça veya nesnenin aktarılması için elektrik enerjisinden faydalanılmasına imkan sağlayan bir düzen oluşturulabilmektedir. Bu sayede hem enerji tasarrufu hem de çevre dostu bir yapı ortaya çıkarılabilmektedir [36].

2.1.4. Makineler arası iletişim modeli

"Makineden Makineye" teriminin İngilizce karşılığının kısaltması olarak M2M terimi kullanılmaktadır. Bu sistem, farklı konumlarda bulunan makinelerin birbirleri arasında veri aktarımı yapabilmelerine imkan sağlamaktadır. Ayrıca M2M sistemi, verilerin biyolojik ve fiziki ortamlar arasında iletilmesine olanak sağlayan uygulamaları da kullanmaktadır [37].

M2M, zeki fabrika sistemlerinde temel ekipmanların yapıya entegre edilmesinde kullanılan bir teknolojidir. Bu şekilde üretim sistemi, algılama, ara bağlantı sağlama ve verilerin entegrasyonunu gerçekleştirme yeteneğine sahip olur. Verilerin analiz edilmesi ve bilimsel karar verme süreçleri, zeki fabrikalar bünyesinde, üretim planlanmasını, ekipmanların verimli kullanımını ve kalite kontrol süreçlerinde kullanılmaktadır. Ayrıca sistem verilerinin yerel bir veri tabanından, bulut sistemine yüklenmesi için internet de ciddi anlamda kullanılmaktadır. Zeki fabrikalar, insan ve makinenin etkileşimi sayesinde, küresel iş birliğine dayalı zeki üretim sistemlerini inşa etmektedirler [38].

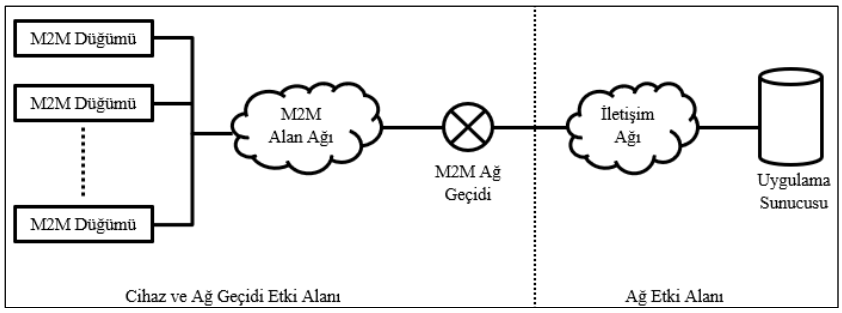
M2M iletişim sistemi, nesnelerin interneti mimarisi içerisinde kullanılan bir veri etkileşim metodudur. Yani, insan müdahalesi olmadan, bilgi alışverişi yapabilen, zeki makine yapısıdır. M2M, nesnelerin interneti mimarisinin anahtar teknolojilerinden birisi konumundadır [39]. M2M, işletme süreçlerinin yönetilmesi için sensörler ile diğer cihazların birbirlerine kablolu veya kablosuz bağlandıkları, kökeni Merkezi Denetim ve Veri Toplama (SCADA) sistemlerine dayanan yeni bir teknolojidir. Bu teknolojide ağa bağlı zeki cihazlar ile insandan bağımsız süreç kararları makineler tarafından alınabilmektedir [40].

M2M, temel anlamda zeki cihazlar, gömülü sistemler, sensörler ve diğer mobil cihaz ve platformlar arasında meydana gelen iletişim ve etkileşimden bahsetmektedir. Bugünün endüstrisinde, sensörlerin ve iletişim teknolojilerinin gelişmesi ile birlikte makineler, bir hayli komplike bir şekle bürünmüş ve endüstri içerisindeki süreçler, sistemlerin kontrolü ve izlenmesi, çok fazla karmaşık ve zor olmuştur. M2M iletişim sistemlerinin kullanılması

ile birçok işlemin izlenmesi ve kontrol edilmesi süreçleri kolaylaşmıştır. Kablosuz sensör sistemlerini kullanarak, uzak birimler kolaylıkla merkezi kontrol birimine bağlanılabilir. Bulut teknolojilerinin gelişmesi ile birlikte sensörler ve zeki cihazlar tarafından üretilen büyük miktarlardaki verinin paylaşılması, depolanması ve analiz işlemlerinin performansı artırılabilir. Bu yüzden, M2M iletişim sistemlerinin popülerliği artış göstermekte ve endüstri içerisinde kullanılmaktadır [41].

Gelişen endüstriyel internetin, işletmelerin süreçlerinde ileriye dönük önemli adımların atılmasına destek sağladığı söylenebilir. Gelişen teknoloji sayesinde yeni nesil iş modellerinin temeli M2M sistemler tarafından oluşturulmaktadır. İşletme içerisinde çalışan makinelerin birbirleri ile bağlantılı duruma gelerek iletişim kurabilmeleri, işletme içi tüm süreçlerin uygulanma şekillerini yeniden tanımlamaktadır. M2M sistemleri sayesinde işletmeler, çok hızlı ve etkin kararlar alabilecek ve kullanılan otomasyon önemi daha da artacaktır. Ayrıca bu sistem nesnelerin interneti kavramını da destekleyerek tüketici cihazları ve hizmetlerini de sistem içerisine dahil edebilecektir [34].

Temel anlamda M2M sistemin yapısı üç katmandan oluşmaktadır. Birinci katman, M2M sistemi içerisinde çalışan cihazlar için veriyi temin eden sunuculardır. İkinci katman iletişim katmanı olarak bilinir ve verinin transferi süreçlerinden sorumludur. Üçüncü katman ise sistemi içerisindeki cihazlardan gelen verilerin işlenmesi süreçlerinin gerçekleştirildiği uygulama katmanıdır [42]. Avrupa Telekomünikasyon Standartları Enstitüsüne göre M2M yapısı, Şekil 2.3'te gösterilmektedir. M2M sistemi, cihaz, ağ geçidi alanı ve ağ etki alanı olmak üzere 3 temel bileşeni içerisinde bulundurmaktadır. Cihaz ve ağ geçidi alanı bileşenleri, M2M sistemleri içerisinde kullanılan çeşitli zeki cihazlardan oluşmaktadır. M2M alan ağı, sistem içerisindeki veri iletişimi için üç tip ortam sağlamakla görevli olan bileşendir. Bu ortamlar; cihazlar arası iletişim, ağ geçitleri arası iletişim ve cihaz ile ağ geçidi arası iletişim olarak gruplandırılmaktadırlar. M2M sisteminin son temel bileşeni olan ağ etki alanı ise verinin, M2M cihazları tarafından internet altyapısı ile yine M2M uygulama sunucularına ulaştırılması ile ilgili görevleri yerine getirmektedir [43].



Şekil 2.3. M2M yapısı [43]

M2M sistemi ile içerisinde sadece makinelerin bulunduğu ortamlar geliştirilerek, bu makinelerin aralarında veri transferlerinin yapılabilmesi için gerekli iletişim ortamı oluşturulabilmektedir. Bu sayede makineler, insanlar olmadan süreçleri gerçekleştirebilmektedirler. M2M teknolojileri sayesinde, ileriki süreçlerde çok daha hızlı işlemlerin gerçekleştirilmesi ve çok büyük miktarlarda verinin üretilmesi ve kullanılması sağlanabilecektir. Sistem, her ne kadar insandan bağımsız olarak düşünülse de sistem içerisinde üretilen verilerin yönetimlerinin sağlanabilmesi için de özel platformlar geliştirilmektedir. M2M yapısı, iş dünyasının kullandığı teknolojinin daha da gelişerek süreçlerin kolaylaştırılmasına ve verimliliğin çok daha iyi seviyelere ulaşmasına imkan sağlayabilmektedir. Ayrıca M2M, bu özellikleri ile Nesnelerin İnterneti kavramının da omurgasını oluşturmaktadır [44].

M2M sistemler, üretkenliği ve verimliliği artırmasının yanında maliyetleri düşürmesinden dolayı, son yıllarda insanlar ve işletmeler tarafından ilgi çekici duruma gelmiştir. Hatta gelecek ağ sistemlerinin dikkat çeken ana konularından biri olarak değerlendirilmektedir. M2M sistemlerin ilgi çeken bir başka konusu ise güvenlik karakteristiğidir. Bu sistemlerin kendisi çeşitli siber saldırılara karşı açıklara sahip olduğundan dolayı, geleneksel siber güvenlik yöntemleri ile korunmaları çok daha zordur [45].

2.1.5. Yapay zeka

İnsanlar, belirli bir zekaya sahip olarak dünya hayatına başlamaktadırlar. Sahip olunan bu zeka yaşantı, çalışma becerileri, öğretim ve eğitim süreçleri vasıtasıyla elde edilen bilgi ve tecrübeler ile daha geliştirilebilmektedir. İnsanın hayatında ilk defa karşılaşabileceği veya aniden karşı karşıya kalabileceği bir durumla ilgili uyum sağlama, problemi anlama, analiz edebilme ve sorun çözüm süreçlerini yönetebilmesi, dikkatin ve odağın gerekli olan yönde

yoğunlaştırılması, zeka ile gerçekleştirilebilmektedir. İnsanların sahip oldukları bu zeka kavramı, teknolojik gelişimler sonucunda bazı donanım ve özel yazılım teknikleri ile makineler tarafından taklit edilebilmektedir. Buna da yapay zeka adı verilmektedir [46].

Yapay zeka, bilgi teknolojileri temeline dayalı olarak çalışan bir makinenin, genel anlamda insanlara atfedilen özellikler olan akıl yürütme, anlama, karar verme, sonuç çıkartma, genelleme yapma ve geçmiş tecrübelerden yeni bilgiler elde etme gibi süreçleri gerçekleştirebilmesi olarak tanımlanabilmektedir. Yapay zeka, zeki programların geliştirilmesini amaçlayan bir bilim dalıdır. Geliştirilmesi amaçlanan zeki programların da yapay zeka statüsüne erişebilmeleri için aşağıda sıralanmış eylemleri gerçekleştirmeleri gerekmektedir [47]:

- İnsanların düşünme becerilerini taklit etmek ve karmaşık problemlerin çözümünü gerçekleştirmek,
- Karşılaştıkları olaylar karşısında yorumlama yapabilmek ve yetkili kişilere sorun hakkında cevaplar verebilmek,
- Öğrenme yeteneğini taklit ederek, çalıştığı alan hakkında uzmanlığını geliştirmek ve sahip olduğu eski bilgileri ile yeni bilgileri uyumlu bir şekilde kullanarak bilgi haznesini geliştirmek.

Yapay zeka çalışmaları, insan zekasını bilgisayar programları vasıtasıyla taklit etmeyi hedeflemektedir. Bu sebeple, çalışmalar kapsamında dil işleme, görüntü analizleri ve süreç planlama eylemler gerçekleştirilmektedir. Yapay zeka, bir bilim dalı olarak insanlardan bağımsız makineler oluşturarak, otonom sistemlerin yaygınlaşması için araştırmalar yapılmasına imkan tanımaktadır. Buradaki ulaşılması istenilen noktalardan biri, makinelerin düşünmesini, anlamasını ve harekete geçmesini sağlamaktır [48].

Yapay zeka, makinelerin daha iyi karar verebilmesi için sistemlere zekilik sağlamakla ilgilenen bir kavramdır. Makine öğrenmesinin daha verimli metodolojiler ve modeller kullanması, sistemlerde yapay zekanın daha da geliştirilmesine imkan sağlamaktadır [49]. Yapay zeka, veri yoğun olarak nitelendirilebilen yaklaşımlardan oluşmaktadır. Sistemlerin çalışması neticesinde elde edilen verilerin, sistemi daha zeki bir hale getirmesi amacıyla kullanılmaktadır. Yapay zeka teknikleri, sistemler içerisinde ortaya çıkabilecek olan hata ve arızaların tespit edilmesinde de kullanılabilmektedir [50].

Yapay zekanın amacının zekiymiş gibi davranan makineler geliştirmek olduğu yönünde ilk defa 1955 yılında John MacCarhty tarafından açıklama yapılmıştır [51]. Yapay zeka üzerine çalışan sistem tasarımcıları, genel bir zekayı modellemek yerine, belirli ölçüde insanların bilişsel/zihinsel yeteneklerini simüle eden ve bilgi temsili gerçekleştiren modellere odaklanmaktadırlar. Yapılan çalışmalar neticesinde günümüzde ortaya çıkmış olan yapay zeka uygulama alanları, algı ve örüntü tanıma, bilgi temsili, problem çözme, akıl yürütme, karar verme, planlama, doğal dil işleme, öğrenme, manipülasyon ve hareket, sosyal zeka, duygusal zeka ve yaratıcılık olarak sıralanabilmektedir [52].

2.1.6. Makine öğrenmesi

Disiplinler arası bir yapıya sahip olan makine öğrenmesi kavramı, matematik, istatistik, bilgi teorisi ve optimizasyon ile ilgili ortak alanlara sahiptir. Makine öğrenmesindeki amaç, makinelerin öğrenebilecekleri şekilde programlanmaları olduğu için bilişim sistemlerini de temel almaktadır. Diğer bir deyişle makine öğrenmesi, yapay zekanın bir dalı konumunda bulunmaktadır. Çünkü, elde edilen tecrübenin uzmanlığa doğru evrimleşmesi veya karmaşık yapılardaki çok fazla veri içerisinden anlamlı örüntülerin ve bilgilerin tespit edilebilmesi gibi insan zekasının temel özelliklerini taklit etmeye çalışmaktadır. Ancak makine öğrenmesinin bunların birkaç adım ilerisinde bulunan zor ve karmaşık görevleri de yerine getirebilme yeteneğine sahiptir. Örnek olarak, yüksek boyutlu veri setleri içerisinde tarama ve analizler gerçekleştirerek, insanların kendi başlarına algılayamayacakları görevleri gerçekleştirebilir ve bu veri setleri içerisindeki örüntüleri ve bilgileri ortaya çıkartabilir [53].

Günümüz şartlarında, teknolojinin birçok alana yapmış olduğu etkiden dolayı devasa boyutlarda veri ve bilgi üretimi gerçekleştirilmektedir. Bu sebeple çok yüksek boyutlarda veri ve bilgi üretimi gerçekleştiren bu sistemlerde de düzensizlik seviyelerinin ölçülmesi gerekliliği ortaya çıkmaktadır. İşte bu noktada öne çıkan kavram, entropidir [54]. Entropi, bilgi teorisi, fizik ve sosyal bilimler alanlarında çok kriterli karar verme problemlerinde ağırlıkların belirlenebilmesi amacıyla kullanılan önemli bir kavramdır. Makine öğrenmesi uygulamalarında bilgi teorisinde entropi, ilk olarak 1948 yılında Shannon tarafından, rastgele bir değişkenle belirsizliğin ölçülmesi olarak tanımlanmıştır [55] [56]. Makine

öğrenmesi uygulamalarında kullanılan bilgi teorisini açıklayabilmek adına seçilebilecek entropi türevleri şunlardır [54]:

- Çapraz Entropi: Sınıflandırma işlemlerinde, tahmin edilen sınıf etiketi ile gerçekte olması gereken sınıf etiketlerinin karşılaştırılması ile gerçekleşen olayı kodlayabilmek için gereken bit sayısının hesaplaması amacıyla kullanılmaktadır.
- Maksimum Entropi: Mevcut veri setinin analizi ile en uygun öğrenme modelinin tespit edilmesi için en uygun veri dağılımının tespit edilmesinde kullanılır. En yüksek entropi değerine sahip veri dağılımın en iyi modeli oluşturduğu kabul edilmektedir.
- Bilgi Kazanımı: Karar ağaçlarının oluşturulması aşamasında, en iyi bölen niteliği tespit edebilmek için kullanılmaktadır. Ağacın kök ve düğümlerinin belirlenebilmesinde kullanılan bu entropi türüne bilgi kazancı adı verilmektedir.
- Kayıp Fonksiyonu: Bu türdeki entropi hesaplamalarını, yapay sinir ağları ile gerçekleştirilen sınıflandırma işlemlerinde, tahmin edilen çıktı ile olması gereken çıktı değeri arasındaki farkı hesaplamak amacıyla kullanılan kayıp fonksiyonları gerçekleştirmektedir.
- Lojistik Regresyon: Temel sınıflandırma türlerinden biridir. Veri seti içerisinde tanımlanan bir olayın meydana gelmesi veya gelmemesi seçeneklerinin oranlarının hesaplanması ile elde edilen sonucun logaritması olarak ifade edilmektedir.
- Kullback Leibler Uzaklığı: Çapraz entropiden farklı olarak, bir olayın gerçekleşme ihtimali ile ikinci bir olayın gerçekleşme ihtimali arasında oluşan düzensizlik değerini hesaplamak için kullanılmaktadır.
- Renyi'nin Entropisi: İstatistik ve ekoloji bilimlerinde genelleştirme değerlerinin hesaplanması için kullanılır. Bu konudaki başarım performansından dolayı, kümeleme tabanlı analiz işlemlerinde kullanılmaktadır.

Makine öğrenmesi kavramı ile ilgili ayrıntılı bilgi aktarımına bu tez çalışmasının dördüncü bölümünde devam edilmiştir.

2.1.7. Büyük veri

Geleneksel verilerin aksine, büyük veri kavramı, heterojen yapıları içeren ve büyüyen, boyutu yüksek veri kümelerini ifade etmektedir. Bu veri kümeleri, yapılandırılmış, yapılandırılmamış veya yarı yapılandırılmış olabilmektedirler. Büyük veri güçlü teknolojiler ve gelişmiş algoritmalar gerektiren karmaşık bir yapıya sahiptir [57].

Emtia donanımlarında depolanması ve işlenmesi konusunda zorluklar yaşanmasına sebep olan 1 Terabyte'tan büyük boyuttaki veri kümesine büyük veri adı verilmektedir. Büyük veri, geleneksel veri işleme ve yönetim teknolojilerinin sınırlamaları nedeniyle pek mümkün olmayan iş farkındalıklarını belirlemek için verileri depolamak, işlemek ve analiz etmek için bir araç ve platform olarak kullanılmaktadır. Ayrıca büyük veri, dağıtılmış ölçeklenebilir alanlarda çok büyük boyutlardaki veri kümelerini işleyebilmek için de kullanılan bir teknolojidir [58].

Furth ve Villanustre, büyük veri kavramının üç temel karakteristiğe sahip olduğunu belirtmiştir. Bunlar [59]:

- Hacim**:** Terabyte seviyesinden Petabyte seviyesine kadar olan veri boyutuna sahip olan, kayıtlar, işlemler, dosyalar ve çizelgeler dahil olmak üzere büyük veri kümelerini ifade eder.
- Hız**:** Toplu, yakın zamanlı, gerçek zamanlı ve akış verileri olmak üzere büyük veri kümelerini aktarma şeklini ifade eder. Ayrıca veri işleme süreçlerinin zaman ve gecikme sürelerini de belirtir. Bu özellik sayesinde veriler, hızlı bir şekilde veya olaylar arasında gerçekleşen gecikme süreleri ile analiz edilebilir, işlenebilir ve muhafaza edilebilir.
- Çeşitlilik**:** Büyük verinin, yapılandırılmış, yapılandırılmamış, yarı yapılandırılmış ve bunların birleşiminden oluşan farklı veri biçimlerini ifade eder. Veri biçimleri, belge, e-posta, metin, resim, video, grafik türlerinde bulunabilir.

Swaminathan ve Palaniswami ise büyük veri hakkında hazırladıkları çalışmada, yukarıdaki belirtilen temel özelliklere, aşağıda listelenmiş olan iki özelliği de eklemişlerdir [60]:

- Değer**:** Veri kümesi içerisinden kullanıcı için elde edilebilecek olan faydayı ifade eder.
- Doğruluk**:** Veri kümesi içerisinde bulunan verilerin, hatasızlığını, güvenirliğini ve kalitesini belirtmektedir.

Günümüz teknolojisinde, büyük veri kaynakları hayatın hemen hemen her alanında bulunmaktadır. İşlemler için kullanılan veriler, radyo frekanslarından, meteorolojik verilerden, sosyal ağlardan, ölçüm cihazlarından, uzaktan algılama sistemlerinden ve konum bilgilerinden elde edilebilmektedir. Gün geçtikçe bu alanlara yenileri eklenmeye devam ettiği için büyük veri konusu da araştırmalarda önemli bir yer kaplamaya başlamıştır. Teknolojinin farklı türlerde cihaz ve hizmetleri kullanabilir ve bunların kullanımından ortaya çıkan verileri toplayabilir hale gelmesinden dolayı kamu, özel sektör ve kişisel kullanım dahil olmak üzere insanın olduğu her yerde büyük veriden söz edilebilmektedir [61].

2.1.8. Sanal gerçeklik

Sanal gerçeklik, bilgisayar ile üç boyutlu olarak hazırlanmış olan tasarımların, kullanıcıların zihinlerinde sanki gerçek dünyada gerçekleşiyormuş gibi hissetmelerini sağlayan dinamik bir araçla, karşılıklı etkileşim ve iletişim halinde sunan bir teknolojidir. Kullanıcılar, sanal gerçeklik sayesinde bulundukları fiziksel ortamdan kendilerini soyutlayarak, o anda sanal ortamda gerçekmiş gibi hareket etmelerine olanak sağlar. Sanal gerçeklik teknolojisi, ilk etaplarda daha etkili oyunlar oynayabilmek amacıyla ortaya çıkmış olsa da günümüz şartlarında sağlık, eğitim, askeri ve üretim alanlarında oluşturulan simülasyonlar ile kendini göstermeye başlamıştır [62].

Sanal gerçeklik, bilgisayarda oluşturulmuş olan üç boyutlu sanal ortamların, birden fazla duyu organına hitap etmesi ile ortaya çıkan etkileşimli simülasyonlardır. Sanal gerçeklik vasıtası ile kullanılan tasarımlar, sanal ortamlar olarak nitelendirilmektedirler. Bu ortamlar, gerçek bir ortamın ya da tamamen hayal ürünü olarak tasarlanan bir ortamın simülasyonu formunda bulunabilirler. Teknoloji sayesinde var olan farklı grafik uygulamaları ile insanların, hayvanların ve nesnelerin sanal ortamlarda gerçeklik hissi ile hareket etmeleri sağlanabilmektedir [63].

Sanal gerçeklik, kullanıcıları için gerçek ortamın bir yansıması şeklinde sentetik bir deneyim sunmayı amaçlayan teknolojidir. Bu çalışma alanı sentetik, yanıltıcı veya sanal olarak adlandırılır, çünkü duyusal uyarım sistem tarafından simüle edilmektedir. Sistem bileşenleri olarak; uyarımı iletmek için çeşitli ekranlar, kullanıcının hareketlerini algılayabilmek için sensörler ve kullanıcı hareketlerini işleyerek çıktıları üreten bilgisayarlar. Geliştiriciler, genellikle kullanıcılar için sanal dünyalar veya sanal ortamlar olarak adlandırılabilen bir bilgisayar modeli üreterek, bileşenler vasıtası ile bunu kullanıcının kullanımına sunarlar. Sanal gerçeklik sistemlerinde başarıya ulaşmanın önemli bir bileşeni, kullanıcının sadece belirli bir duyguyu yaşamasından ziyade, eylemleri ile sanal ortamı şekillendirebilmesi ve ortamı yaşayabilmesidir [64].

2.1.9. Artırılmış gerçeklik

Artırılmış gerçeklik, sanal ortamda tasarlanan nesne ve modellerin, gerçek dünyada insanlar tarafından gerçekten orada varmış gibi kullanılabilmesidir. Bu teknoloji insanlar, yaşadıkları gerçek dünyada, sanal nesneler ile etkileşime geçebilmektedirler. Bu etkileşimin yaşanabilmesi için özel donanımlar aracılığı ile sanal nesneler ve modeller oluşturulabilir. Bu sayede artırılmış gerçeklik günümüzde pazarlama, emlak, reklam, sinema, sanat, oyun, askeri, lojistik ve alışveriş gibi birçok alanda kullanılabilmektedir [62].

Artırılmış gerçeklik, gerçek fiziksel dünya ile dijital olarak oluşturulan sanal dünya arasındaki farkı bulanıklaştırmaktadır. Bu da bilgisayarların gerçek dünyadaki bir sahneye gerçek zamanlı olarak farklı bir görüntü ekleyebilme yeteneğine dayanmaktadır. Temel olarak, Apple ve Google tarafından geliştirilmiş olan haritalar uygulamasında, kullanıcı nereye gidileceği ve nasıl gidileceği ile ilgili sesli komut verip, buna karşılık uygun bir cevap alabiliyorsa bir artırılmış gerçeklik deneyimi yaşıyor demektir. Artırılmış gerçeklik sistemleri, bir LCD veya diğer cam ekranlarda bir görüntü oluşturabilse de bunun yanında ses, dokunsal ve lazer etkileşimi de dahil olmak üzere farklı özellikleri de kullanabiliyorlar. Ayrıca zaman geçtikçe kullanabildikleri özellik sayısı da artmaya devam ediyor [65].

Artırılmış gerçeklik, sanalda oluşturulmuş olan nesnelerin, gerçek dünyadaki bir ortam üzerine eklenerek kullanılabilmesine odaklanan bir sanal gerçeklik varyasyonudur. Artırılmış gerçeklik, uygulamada başarıya ulaşabilmesi için üç gereksinimi karşılamalıdır.

Bu gereksinimler; sanal ve gerçek nesnelerin gerçek bir ortamda birleştirilmesi, aralarında bir etkileşim oluşturulması ve bu etkileşim içerisinde gerçek zamanlı olarak çalıştırılabilmesi olarak ifade edilmektedir. Sanal gerçekliğin, insan duyularına sunduğu sanal ortamı kullanabilmesine karşılık, artırılmış gerçeklik sanal olarak oluşturulmuş nesnelerin üç boyutlu olarak gerçek dünyada var olabilmelerini ve kullanılabilmelerini ifade etmektedir [66].

2.1.10. Bulut bilişim teknolojisi

Bulut bilişim kavramı, işletmelerin otomasyon yazılımını yönetme ve verileri depolama hizmetlerinin uzak sunucularda internet vasıtasıyla gerçekleştirilmesi olarak tanımlanmaktadır. Yani işletme faaliyetlerinde elde edilen tüm veriler, çevrimiçi olarak depolanacak ve internet erişimi olan tüm bilgisayar ve mobil cihazlardan erişim sağlanabilecektir. Bu sayede zaman ve mekandan bağımsız erişim esnekliği ortaya çıkmıştır. Ayrıca bulut bilişim hizmetlerinin web tabanlı uygulamalar ile işletim sistemine herhangi bir yazılım kurmaya gerek kalmadan yönetilebilmesi ve birçok bulut bilişim hizmetinin belirli sınırlar içerisinde ücretsiz olması kullanıcılar için hayatı daha kolay bir hale getirmektedir [32].

Bulut bilişim, kullanıcılara ağ tabanlı bir ortam oluşturarak, konuma bağımlı kalınmadan hesaplamaların ve kaynakların kullanımının önünü açmaktadır. Ulusal Standartlar ve Teknolojiler Enstitüsüne (NIST) göre bulut bilişim, sağlayıcı tarafından çok az bir denetim ve iletişim ile hızlıca serbest kalabilen, programlanabilir araçlar, depolama, sunucular, yazılımlar ve imkanların bir arada bulunduğu bir ortamı kapsayan, her ihtiyaç duyulduğunda internet üzerinden erişilebilir bir tema olarak tanımlanmaktadır. Bulut bilişimin işleme türü özellikleri arasında, yüksek performanslı ağ erişimi, hızlı esneklik, kaynak depolama ve ölçülebilen servisler bulunduğu belirtilmektedir [67].

Su moleküllerinin toplamı olarak ortaya çıkan gerçek bulutlar gibi, bulut bilişim sistemlerindeki bulut terimi de ağların toplamı olarak tanımlanmaktadır. Kullanıcılar, bir bulut bilişim sistemine dahil olduktan sonra, bulut kapsamında bulunan tüm kaynakları istedikleri zaman, konumdan bağımsız bir şekilde kullanabilmektedirler. Kullanıcılar, bulut bilişim kullanımında, genellikle kendilerine ait bir yapı kurmak yerine, aracı olarak bulut hizmeti veren bir sağlayıcı firma tercih etmektedirler. Bulut bilişim tarafından

kullanıcılarına hizmet olarak yazılım (SaaS), hizmet olarak platform (PaaS) ve hizmet olarak altyapı (IaaS) şeklinde üç adet hizmet sunulmaktadır. Ayrıca bulut bilişim üç temel bileşene sahip olmalıdır. Bunlar[68]:

- İstemciler**:** Son kullanıcı, istemci olarak çalışan bilgisayarlar ile bulut bilişime erişim sağlayabilir ve hizmetleri kullanabilir,
- Dağıtılmış sunucular**:** Sunucular, farklı konumlara dağıtılabilir, ancak beraber çalışıyorlarmış gibi davranırlar,
- Veri merkezleri**:** Sunucuların derlenerek oluşturulan, verilerin depolandığı ve işlendiği merkezlerdir.

2.1.11. Simülasyon teknolojileri

Simülasyon teknolojisi, bir modelin sanal bir kopyasını oluşturmak ve gerekli deney ve testlerin model üzerinde yapılmasına olanak sağlamak amacıyla bilgisayar grafikleri ve yazılım teknolojisinin kullanılması olarak tanımlanabilmektedir. Simülasyon teknolojilerinin avantajları, yüksek verimlilik, nispeten yüksek güvenlik seviyesi, çevresel koşullar için daha düşük gereksinimler ve nispeten daha düşük sınırlamalar olarak ifade edilebilir. Simülasyon kapsamında oluşturulan modelin ihtiyaçları, zaman içerisinde yapılan deneyler ve testler ile daha isabetli bir şekilde belirtilebilir [69].

On yıllardır uygulanan simülasyon teknolojileri, insanların teori ve deneyler dışında nesnel dünyayı daha iyi tanımalarına ve geliştirmelerine imkan sağlamaktadır. Simülasyon teknolojileri havacılık, sağlık, malzeme bilgisi, üretim, enerji ve tarım vb. geniş bir çalışma alanı yelpazesine sahiptir. Aynı zamanda savunma sanayinde ulusal güvenliğin daha iyi seviyelere çıkarılabilmesi amacıyla toplum, ekonomi ve askeri alanların tamamında kendisine yer bulabilmektedir. Bu alanlar ile ilgili yapılan araştırmalarla ilgili en önemli ortak nokta, araştırma nesnelerinin son derece karmaşık, belirsiz ve doğrusallıktan uzak olmalarıdır. Örneğin üzerinde çalışılan tek bir sistem bile hem nicel hem nitel hem sürekli hem de ayrık verilere sahip olabilmektedir. Bu tür özellikler, sistem üzerinde geleneksel yöntemler ile teorik ve deneysel çalışmalarını yapılabilmesini önemli ölçüde zorlaştırmaktadır. Bu tür durumlarda araştırmanın gerçekleştirilebilmesi için simülasyon teknolojilerini kullanmanın gerekliliği ortaya çıkmaktadır [70].

Zeki sistemlerin işlevselliklerini uygulayabilmek için simülasyon teknolojilerinin kullanımına ihtiyaç vardır. Tanımlama, ağ kurma, optimize etme, devreye alma gibi olayların gerçekleştirilmeden önce simüle edilmesi, kurulması gereken gerçek sistemin ayrıntılı dijital modellerinin denenebilmesine imkan tanımaktadır. Endüstri 4.0 süreçlerinde simülasyonlar, yalnızca geliştirme uygulamaları için hazırlanmış mühendislik uygulamaları değildir. Simülasyonlar zeki sistemleri, sezgisel kullanıcı ara yüzlerini ve eğitim simülatörlerini tasarlamak için gerçek sistemlerin içerisinde kullanılabilmektedir. Bu sebeple, gerçek sistemlerin yaşam döngüleri boyunca değişen uygulamalı senaryolarında, simülasyon teknolojilerini tutarlı bir şekilde kullanmaları gerekmektedir [71].

Bir sistemin simülasyonunu oluşturabilmek için simülasyon türlerini incelemek gerekmektedir. Bu konuda iki farklı simülasyon türü bulunmaktadır. Bunlar [72] [73]:

- Sürekli Olay Simülasyonu**:** Bu simülasyon türü, zaman içerisinde sürekli olarak değerleri değişen değişkenlere sahiptir. Bu sebeple sürekli olay simülasyonlarında sonuç değerleri, sistemin çalışması kararlı duruma geldikten sonra belirli aralıklarla alınmaktadır. Kimyasal madde üretim sistemleri, gaz boru hattı sistemleri ve sürekli faal bulunan tıbbi malzeme satış noktaları için yapılabilecek projelerde kullanılmaktadır [72]. Diğer bir ifadeyle, sürekli simülasyonlarda durum değişkenleri zaman içerisinde sürekli olarak değişmektedir [73]. Sürekli olay simülasyonlarının bir sistem kapsamında, tanımı gereği yumuşak geçişleri temsil ettiği iyi bilinen bir özelliktir [74].
- Ayrık Olay Simülasyonu**:** Bu simülasyon türü, zaman içerisinde farklı noktalarda değerleri değişebilen değişkenlere sahiptir. Banka faaliyetleri, market satış sistemleri, envanter kontrol sistemleri ve ayrık ürün üretim sistemleri gibi alanlarda uygulanmaktadır [72]. Ayrık olay simülasyonu, zaman içinde meydana gelen bir faaliyetin ayrıntılarını ortaya çıkartmak için olayları kullanmaktadır. Kullanılan ayrık matematiksel analiz yöntemi, sistemi modelleyebilir ve orta seviyede bir soyutlama gerçekleştirebilir. Simülasyon içerisinde bulunan her olay, benzersiz bir kimliğe sahip bir işlev veya sınıf olarak gerçekleşebilir [73]. Ayrık Olay Simülasyonu, gerçek dünyayı mümkün olduğunca temsil edebilmek ve olay bazında dinamikleri simüle edebilmek amacıyla kullanılmaktadır. Güçlü

bilgisayarların kullanım oranlarının artması dolayısıyla, bilgisayar destekli karar verme sistemlerinin önemli bir parçası haline gelmiştir [75].

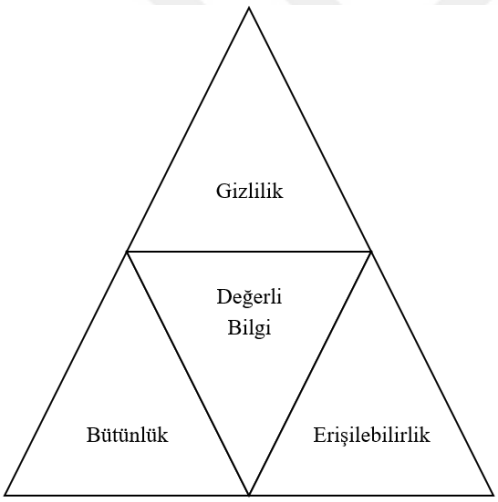
Zeki fabrikalar üzerine yapılan araştırmalarda sanal kurulumların tercih edilmesi ve simülasyonların kurulması, son zamanlarda karşılaşılan bilimsel eğilimlerdendir. Bir çalışma için öncelikli olarak simülasyon kurulması ile iş yükünden tasarruf edilmektedir. Bu sayede sistemdeki hata riski en aza indirilebilir ve sistemin tasarımı ve konfigürasyonu optimum seviyede gerçekleştirilebilir [76].

2.1.12. Siber güvenlik

Siber güvenlik, bir varlık türü olarak bilginin izinsiz veya yetkisiz bir biçimde erişim, kullanım, değiştirilme, ifşa edilme, ortadan kaldırılma, el değiştirme ve hasar verilmesini önlemek olarak tanımlanır [77]. Siber güvenlik, her organizasyonun sürekliliğinin sağlanmasında büyük önem taşır ve organizasyonun başta elektronik olmak üzere, çeşitli ortamlardaki kritik bilgilerinin ve diğer bilgi varlıklarının korunmasını sağlar. Sadece büyük şirketler, holdingler değil bunun yanı sıra KOBİ'ler, devlet kurumları veya kar amacı gütmeyen herhangi bir organizasyon, okul, vb. de siber güvenlik sorunları ve risklerini farklı düzeylerde de olsa sürekli yaşamaktadır. Bu gerçek, dünya genelinde olduğu gibi ülkemizde de sürekli artan boyutlarda ortaya konan bir olgu haline gelmektedir [78].

Siber güvenlik, bilgisayar ve bilgisayar güvenliği kavramları da dahil olmak üzere çok geniş bir çalışma alanı oluşturmaktadır. Bu çalışma alanı, bilgisayar bilimi, işletme yönetimi, bilgi araştırmaları ve mühendislik süreçlerini kapsayan çok disiplinli bir yapıya sahiptir. Hedef olarak bilgi ve verilerin güvenliği olduğu bir kapsam oluşturulmaya çalışılmaktadır. Bu kapsamda bulunan bilgi veya veriler, ya ağ içerisinde sürekli hareket halinde bulunmaktadır ya da bir hafıza biriminde depolanmış durumda muhafaza edilmektedir. Bu sebeple siber güvenlik, yalnızca kriptografik, iletişim, taşıma ve değişim protokollerinin en iyi uygulamalarının ayrıntılı matematiksel tasarımları ile yetinemez. Aynı zamanda hem statik halde bulunan hem de hareket halinde bulunan veri ve bilgilerin durumlarının da analiz edilmesini içermektedir [79].

Siber güvenliğin merkezinde, üç temel güvenlik özelliği bulunmaktadır. Bunlar gizlilik, bütünlük ve erişilebilirlik kavramlarıdır (Şekil 2.4). Gizlilik, bilgilerin yetkisi bulunmayan kişi veya kuruluşlar tarafından erişilebilmesinin engellenmesi olarak ifade edilebilmektedir. Bütünlük, bilgilerin iletim veya muhafaza süreçlerinde doğruluklarının ve eksiksizliklerinin, kasıtlı, tesadüfi veya yetkisiz bir şekilde korunması olarak tanımlanabilir. Erişilebilirlik kavramı ise bilgilerin yetkili kişi ve kuruluşlar tarafından gerekli olunan her an erişilebilmesi olarak belirtilmektedir. Uygulamalar ile azaltılan her risk, yapılan her kontrol, bu özelliklerin perspektifinde gerçekleştirilmektedir. Siber güvenlik bu özellikleri koruyabilmek için güvenlik önlemleri sağlamaktadır. Bu nedenle işletmelerin güvenlik seviyelerini artırabilmek için çeşitli çözümler tasarlarken, bu özellikleri etkileyen tehditlerin analiz edilmesi gerekmektedir [80].



Şekil 2.4. Siber güvenliğin üç ilkesi [80].

Siber güvenlik yaygın olarak hem resmi hem de sivil alanlarda birbirine bağlı ağları, bilgi alt yapılarını, bunlara zarar verebilecek tüm tehditlerden korumak için alınmış olan mevcut önlemler ve eylemler anlamına gelmektedir. Siber güvenlik, ağların ve altyapının erişilebilirliğini, bütünlüğünü ve içerisinde bulunan bilgilerin gizliliğini korumaya çalışmaktadır [81].

2.2. Endüstri 4.0 ve Güvenlik Endişeleri

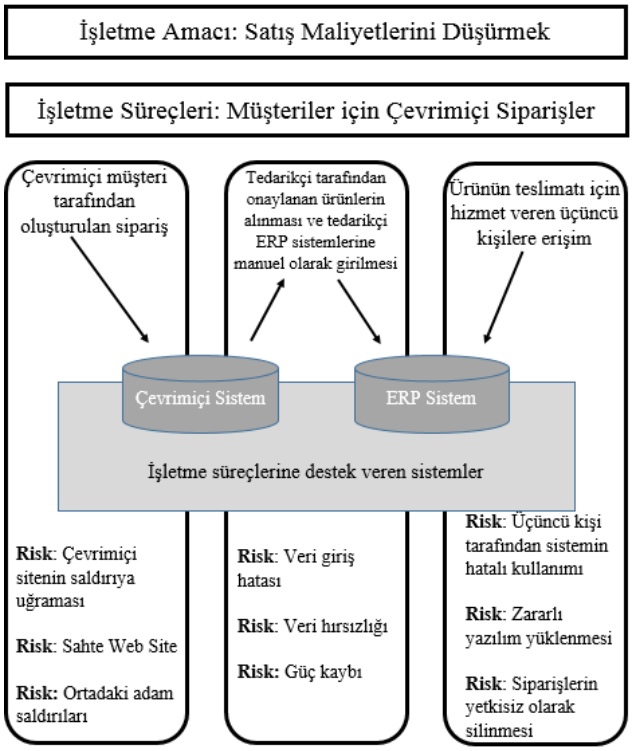
Nesnelerin interneti ve M2M sistemleri, tüm zeki cihazları birbirine bağlamayı amaçlayan teknolojik yapılar olarak karşımıza çıkmaktadırlar. Bu yapılar, kişisel sistemler, endüstriyel otomasyonlar ve tamamen özerk sistemlere kadar her şeyin mümkün olabileceği bir süreci

önemli ölçüde iyileştirme potansiyeline sahiptir. Nesnelerin interneti ve M2M sistemlerinin endüstriyel değer zincirine entegrasyonu, üretim sistemlerinin bilgisayarlaşmasını teşvik etmek amacıyla "Endüstri 4.0" olarak adlandırılmaktadır. Endüstriyel anlamda bu yapıların entegre edilmesi işletmelere birçok anlamda maliyetlerin düşürülmesini vaat etmektedir. Bunlarla birlikte bu yeni gerçeklik, bu yapıların güvenli bir şekilde oluşturulması konusunda önemli zorluklar ortaya çıkarmaktadır. Sistem içerisinde kullanılacak cihazların bilişim sistemleri alt yapısı ile kendi aralarında ve internet ile bağlantılı olacakları düşünüldüğünden güvenlik ve hatasızlık kavramı, nesnelerin interneti ve M2M sistemlerinin en büyük endişelerinden biri olarak gösterilmektedir [82].

Fiziksel cihazlardan oluşan bilgi ağı sistemlerinde yaşanan artış, sistemleri yaşam süreleri boyunca güvenlikle ilgili sorun yaşamalarına sebep olmaktadır. Veriye erişim imkanı sağlayan sensörler ve zeki cihazlar tarafından üretilen, iletilen ve depolanan bilginin yönetimindeki zorluk seviyesi, ağın genişlemesi ile daha da yükselmeye devam etmektedir. Bilgi ağı içerisinde mevcut olan verilerin, dinleme, değiştirme ve yok etme gibi birçok saldırıya maruz kalma ihtimalleri bulunmaktadır. Bu sebeplerden dolayı, işletmelerin özellikle bütünlük konusunda endişeleri ortaya çıkmaktadır. Konu hakkında yapılan çalışmalar, özellikle M2M alanında verinin yönetiminin ve bütünlüğünün sağlanmasının zorluğunu ortaya koymakta ve bu alanda sürekli olarak çalışmalar yapılmasına devam edilmesi gerekliliğini vurgulamaktadır. Çünkü siber fiziksel sistemler, kullanım amaçları itibari ile kabul edilemez güvenlik risklerine sahiptir [83].

Endüstri 4.0 devrimi ile üretim sistemindeki cihazların bilgisayar ağı altyapısı ile birbirlerine bağlanmaları ve veri alışverişinde bulunmaları amacıyla kullanılan M2M sistemleri, mevcut olan tüm güvenlik tehlikeleri ile başa çıkmak zorunda kalmıştır. M2M sistemler, internet dünyasına yeni tehditler kazandırmamıştır, ancak var olan tehditlerin etki alanının ve vereceği zararın boyutunu artırmıştır [84]. Veri gizliliği, kimlik doğrulama, bütünlük, reddedilme ve kullanılabilirlik kavramları, güvenlik sistemleri içerisinde değerlendirilmesi gereken ana konular olarak göze çarpmaktadır. Ancak, M2M sistemler için bu güvenlik sistemlerinin kullanılması hakkında bir standart bulunmamaktadır. M2M sistemler içerisinde bulunan zeki makineler ve bilgisayarlar, internet tabanlı daimi iletişim içerisinde bulunduklarından dolayı çeşitli aktif ve pasif saldırılara karşı savunmasız durumdadır [45].

Teknolojik gelişmeler, işletmelerin verimliliklerini ve performanslarını artırabilmeleri için günlük iş süreçlerinde eşi benzeri görülmemiş bir baskı oluşturmuştur. Gerçekte işletmeler, yeni teknolojilerin ortaya çıkması ile birlikte daha kolay iş süreçlerini ortaya çıkartan işletme aktivitelerini geliştirmek için bilgi ve iletişim teknolojilerinin performansına ciddi anlamda bel bağlamaktadırlar. Bulut tabanlı sistemler, nesnelerin interneti ve endüstri 4.0, teknolojik değişimler konusunda yeni girişimler başlatmış fakat bunlarla beraber yeni güvenlik risklerinde de artışa sebep olmuştur. Birbirleri ile bağlantılı olan sistemler işletmelerin, kritik dereceye ve finansal etkilere sahip birçok güvenlik riskine maruz kalma oranını önemli ölçüde artırmaktadır [85].



Şekil 2.5. İşletme süreçlerinde karşılaşılabilecek bazı güvenlik riskleri [85]

Siber fiziksel sistemler, genel manada üç yapıdan oluşmaktadır. Bunlar; bilgisayar sistemi, fiziksel sistem ve bu ikisini birbirine bağlayan iletişim sistemidir. Bilgisayar ve iletişim sistemi birlikte endüstriyel kontrol sistemlerinin oluşmasını sağlamaktadır. Bu endüstriyel kontrol sistemleri, son zamanlarda üretim sistemleri de dahil olmak üzere birçok yapı içerisinde kullanılmaktadır. Siber fiziksel sistemler genel anlamda izole edilmiş ağlar şeklinde kurulmalarına rağmen, bazı sistemler, ekonomik ve operasyonel faydalarından dolayı, kamusal iletişim ağının (İnternet) kullanılması ile fiziksel üretim sistemlerinden coğrafi olarak farklı konumlarda kurulması yoluna da gidilmektedir. Ancak kamusal

iletişim ağının kullanılması, faydaları ile beraber siber güvenlik açıklarını ve olası saldırı ihtimallerini de beraberinde getirmektedir. Bu sebeple, kamu iletişim ağı alt yapısı ile kullanılacak şekilde kurulan zeki fabrikalara, siber saldırılar sebebiyle oluşması muhtemel zararları en aza indirmek için, saldırıları gerçek zamanlı tespit etme yeteneği kazandırılmalıdır. Bu amaçla, günümüzde yapılan çalışmaların birçoğunda anomali tespiti üzerine odaklanılmıştır. Anomali tespit yapılarının bu amaçla kullanılması ile oluşabilecek herhangi bir saldırının, hangi cihaza yönelik konum bilgisi ile anında tespit edilmesi, en az zarar ile engellenmesi, hedeflenmektedir [86].

3. ANOMALİ VE ANOMALİ TESPİTİ

Bu bölüm altında, anomali kaavramının tanımı ile birlikte çeşitleri ile ilgili bilgiler verilmiştir. Sonrasında da anomali tespit sistemlerinde kullanılabilecek yaklaşımlar hakkında ayrıntılar aktarılmıştır.

3.1. Anomali

Anomali, normalden önemli derecede farklılık gösteren verilerdir. Birçok araştırma çalışması, doğada belirli kuralları ve geniş ilkeleri izleyen, gözlemlenebilir bir sistemin durumuyla sonuçlanan süreçlerin ve davranışların var olduğu varsayımına dayanmaktadır. Bu normal süreç davranışları neticesinde de normal değer aralıklarına sahip veriler elde edilebilmektedir. Ancak süreçlerde normalden sapmalar meydana gelebilir ve sistemler anormal durumlarda da çalışmaya devam ederler. Süreçlerde yaşanabilecek olan bu sapmalar sonucunda ortaya çıkacak veriler de normal değer aralıklarından önemli derecede farklılık göstereceklerdir. Bu veriler anomali olarak tanımlanmaktadır [5].

Anomaliler, normal süreçler içerisinde çeşitli anormal faaliyetler sebebiyle ortaya çıkmaktadırlar. Anomaliler, Nokta Anomalisi, Bağlamsal Anomali ve Toplu Anomali olarak sınıflandırılır [6].

3.1.1. Nokta anomalisi

Nokta anomalisi, bir veri örüntüsü, veri setinde bulunan aynı türdeki diğer tüm verilerden farklı davranış gösteriyorsa ortaya çıkmaktadır. Yaygın olarak karşılaşılan anomali türüdür [87].

Tek bir veri örneği, birlikte bulunduğu diğer tüm veriler için anormal olarak tanımlanabiliyorsa, nokta anomalisi olarak isimlendirilir. Nokta anomalisi en basit anomali formu olarak bilinmektedir [88].

3.1.2. Bağlamsal anomali

Bağlamsal anomali, bağlamsal özellikler ile birlikte davranışsal özellikleri de kapsayan veri setleri içerisinde bulunurlar [89]. Örneğin bir veri belirli dönem veya ortamda normal davranış sergilemesine rağmen, aynı veri farklı bir dönem veya ortamda anomali olarak değerlendirilebilir [90].

Bağlamsal anomaliler iki niteliğe sahiptirler. Bunlar, bağlamsal nitelikler ve davranışsal niteliklerdir. Bağlamsal nitelikler, bir veri örneğinin bağlamını belirlemek amacıyla kullanılmaktadır. Örneğin, herhangi bir konumun enlem ve boylam bilgileri, ait olduğu veri seti için bağlamsal nitelik olarak değerlendirilmektedir. Davranışsal nitelikler, bir örneğin bağlamsal niteliklerinin tanımlanamadığı yerlerde kullanılmaktadırlar. Örnek olarak dünya üzerinde meydana gelen ortalama yağış miktarını tanımlayan bir veri kümesinde, herhangi bir bölgede meydana gelen yağış miktarı davranışsal niteliği oluşturmaktadır [88].

3.1.3. Toplu anomali

Toplu anomali, belirli bir veri örüntüsü, tüm veri kümesine göre aykırı davranış sergiliyorsa ortaya çıkan anomali türüdür. Örneğin bir veri seti içerisinde toplu halde bulunan anomali verileri bireysel olarak anomali oluşturmayabilir. Ancak bunların birlikte değerlendirilmeleri, toplu anomaliyi işaret eder [91].

Son zamanlarda dijital işleme, sosyal ağlar gibi alanlarda toplu anomali tespitine yönelik uygulamalar önem kazanmaya başlamıştır. Toplu anomalilerin yapısı sebebiyle, sistemin yanlış alarm verme oranı yüksek olduğu için anomali verileri ile normal verileri birbirlerinden ayırt edebilmek önemli bir konudur. Yüksek boyutlu veri setleri içerisinde toplu anomali oluşturan veri örüntülerinin, bir öğrenme ve eğitim modeli kullanmadan tespiti zordur. Bilinen denetimsiz anomali tespit yöntemleri, veri seti içerisinde bulunan anomali verilerinin oranının, bütüne göre çok düşük olduğunu varsaydığından dolayı, başarı performansları toplu anomali tespiti için pek tatminkar değildir [92].

3.2. Anomali Tespiti

Anomali tespiti, bir sistem içerisindeki anormallikleri, hataları ve kusurları ortadan kaldırmak ve sistemin kalitesini yükseltme amacıyla gerçekleştirilen bir ortaya çıkarma sürecidir [93]. Anomali tespiti, genel anlamda belirli bir veri seti içerisinde istenen şekle ve özelliklere uymayan veri, veri grupları ve örüntüler tespit edilmesi ve tanımlanmasıdır [94]. Farklı bir tanıma göre anomali tespiti, normal eylemler sırasında ortaya çıkan ve veri kümesi içerisinde normalin dışında bir davranış sergileyen kalıpları bulma sürecidir. Bu kalıplar, anomali veya sapan veri olarak değerlendirilebilir [7]. Veri setleri içerisinde gerçek zamanlı anomali tespitinin yapılabilmesi, ortaya çıkabilecek olan arızaların minimum sürede belirlenerek çözüm üretilmesi için gerekli bir eylemdir [95]. Anomali tespiti, veri seti içerisinde bulunan bir değerin önceden belirlenmiş bir sınır değerinin aşılmasının tespitinden ziyade, veriler arasındaki bağımlılıkları otonom olarak belirleyen ve bunları sürekli izleyerek analiz eden bir yapıdır [96].

Anomali tespit sistemleri, teorik olarak sağlam temellere dayanmaktadır ve üretim sistemlerinde ortaya çıkabilecek küçük, orta veya büyük ölçekli sorunlar için hızlı tespit, kolay bakım ve yeniden kullanılabilirlik desteklenmektedir. Bu sayede geliştirilmesi planlanan modellerin erken test süreçlerine tabi tutulmalarını sağlarlar. Anomali tespit sistemleri hem denetimli hem de denetimsiz sistemlerde kullanılabilirler [76]. Anomali tespit sistemlerinin en büyük faydalarından biri, gerçekleşme ihtimali olan yeni hataları da tespit edebilmeleridir [97].

Anomali tespit tekniklerinin kullanım alanları şu şekilde listelenebilir [5] [98]:

- Bilgilerin gizliliğinin sağlanması, zararlı yazılımların tespit edilmesi ve sahte e-postaların engellenmesi ve ayrıca sistemlere yetkisiz girişlerin saldırı tespit sistemleri ile zamanında ortaya çıkabilmesi amacıyla bilgi güvenliği alanında kullanılmaktadır.
- Kredi kartı dolandırıcılıklarının tespit edilebilmesi, müşterilerin kredi geri ödeme tahminlerinin yapılabilmesi, işletmelerin iflas tahmin edilebilmesi ve yatırım araçlarındaki olumsuzlukların önceden belirlenebilmesi için finans alanında kullanılmaktadır.

- Hastalıkların doğru teşhis edilebilmesi, hastanın takip edilmesi, radyoloji görüntülerinden başarılı sonuçlar çıkarılabilmesi ve salgın hastalıklarla ilgili sağlıklı tahminler yapılabilmesi için sağlık alanında kullanılmaktadır.
- Personel davranışlarının, savaş alanı davranışlarının ve beklenmeyen saldırı ihtimallerinin başarılı bir şekilde değerlendirilebilmeleri için savunma ve iç güvenlik sistemlerinde kullanılmaktadır.
- Ev içi kazaların tespit edilebilmesi, evlerin çevre güvenliklerinin takip edilebilmesi ve iç mekanlarda olabilecek sağlıksız koşulların belirlenebilmesi amacıyla tüketici güvenliğinin sağlanması alanında kullanılmaktadır.
- Kalite kontrol işlemlerinin daha başarılı yapılabilmesi, işletmelerin başarılı stok ve envanter yönetimi gerçekleştirebilmesi, işletmeler için müşteri ve çalışan davranışlarında yaşanabilecek olumsuzlukların başarılı bir şekilde tespit edilmesi amacıyla işletme alanında kullanılmaktadır.
- Üretim sistemlerinde bulunan cihazlarda, sürekli çalışmadan dolayı zamanla oluşabilecek aşınma ve yıpranma gibi nedenlerle veya yapısal kusurlar sebebiyle oluşabilecek mekanik veya elektronik anormalliklerin gerçek zamanlı tespit edilebilmesi için üretim ve endüstri alanında kullanılmaktadır.
- Banka ve sigorta şirketleri gibi çeşitli işletmelerde gerçekleşebilecek suç faaliyetleri ve kötü niyetli müşterilerin eylemlerinin zamanında tespit edilebilmesi amacıyla dolandırıcılık tespiti işlemlerinde kullanılmaktadır.
- Bir bakış açısında göre bilimdeki ilerlemenin, normal düzene uyum sağlayamayan anormal verilerin keşfedilmesi ile ortaya çıkan paradigmatik devrimler ile gerçekleşmektedir. Bu sebeple anomali tespiti, bilimsel çalışmaların her alanında yer almaktadır.

Tüm alanlarda anomali tespit tekniklerinin kullanılmasında, istatistiksel, makine öğrenmesi ve sonlu durum makineleri yöntemleri kullanılmaktadır. İstatistiksel yöntemlerde, elde edilen veriler üzerinde çeşitli hesaplamalar yapılarak geçerli bir profil oluşturulur. Bu profil içerisinde ortaya çıkan eşik değerini aşan veriler anomali olarak değerlendirilir [99]. Makine öğrenmesi yöntemine dayalı yaklaşımlar, veri setini oluşturan ve yapı içerisindeki normal düzenden sapan verileri anomali olarak değerlendiren bir faaliyet profili olarak çalışmaktadır [100]. Sonlu durum makineleri yönteminde ise sistemlerin sürdürülebilirlik şartlarının bozulmasına neden olan anomalilerin tespit edilmesi sağlanabilir [101].

Anomali tespit işlemi, siber fiziksel sistemler içerisinde gerçekleştirilen süreçlerin güvenliğinin sağlanması için son yıllarda uygulanmaya çalışan konulardan biridir. Anomali tespiti, veri setleri içerisinde oluşabilecek olan anormal veri kalıplarını ortaya çıkartmaya odaklanmaktadır. Veri setinin büyük bir kısmından ayrılan, farklı bir görüntüye sahip olan örüntüler anomali olarak nitelendirilmektedir ve bu tür örüntüler sistemin işleyişi konusunda ciddi problemlere yol açabilmektedirler. Günümüzde anomali algılama yöntemleri, dolandırıcılık tespiti, izinsiz giriş tespiti, anormal görüntü tespiti, tıbbi teşhis ve üretimde hatalı eylem tespiti gibi birçok süreç için kullanılabilmektedir [102].

Anomali tespiti, sistem içerisinde akan verilerin barındırabileceği anormallikler ve istisnaları tespit etmek amacıyla kullanılan bir yöntemdir. Genel anlamda makine öğreniminin sınıflandırma algoritmaları kullanılarak, iki seçeneğe sahip sınıf etiketlerinin eklenmesi ile gerçekleştirilmektedir. Eklenen sınıf etiketi, ilgili veri satırının anomali barındırıp barındırmadığı konusunda bilgi vermektedir. Makine öğrenmesi yaklaşımının kullanıldığı, denetimli ve denetimsiz olmak üzere iki farklı anomali tespit yöntemi bulunmaktadır. Denetimli anomali tespit yöntemi hem normal verilere hem de anormal verilere ait mevcut olan etiketli örnekler ile gerçekleştirilir. Bu yapının en iyi avantajı, sistem anomali verilerini başta oluşturulan bir model ile öğrendiği için, yüksek doğruluk oranı ve yüksek işlem hızı göstermektedir. En kritik dezavantajı ise, etiketler ile modele öğretilmemiş ancak olması muhtemel anomali türleridir. Denetimsiz öğrenme modelinde ise, çeşitli normal ve anormal veriler için bir dizi varsayım oluşturulmaktadır. Denetimli sisteme göre daha düşük doğruluk oranı ve daha düşük işlem hızı vermesine rağmen, öngörülmemiş anomalileri de yakalama imkanından dolayı tercih sebebi olabilmektedir [103].

Zeki üretim sistemlerini içerisinde barındıran siber fiziksel sistemler için son zamanlarda ortaya çıkan görevlerden biri üretim hatlarında gerçek zamanlı oluşabilecek olan anomalilerin hızlı ve doğru bir şekilde tespitinin yapılabilmesidir. Geçmiş zamanlarda bu görev, üretim hattında sürekli olarak sistemi gözleyen denetçiler tarafından yapılamaya çalışılırken, siber fiziksel sistemler içerisinde artık tamamen sensörler tarafından okunan verilerin gerçek zamanlı olarak kontrolden geçirilerek hata olup olmadığının tespit edilmesi ve tespiti ekranlar karşısında takip etmeye çalışan denetçilerin bulunması şeklinde yerini almaya başlamıştır. Bu sayede üretim hatlarında kullanılan cihazların ciddi hasarlar

alması ve hatalar dolayısıyla oluşması muhtemel ekonomik kayıpların önüne geçilmesi planlanmaktadır. İlerleyen süreçte anomali tespit sürecinin tamamen otonom bir yapıya kavuşturulması ve insandan bağımsız bir şekilde gerekli sistemlerin tetiklenmesini sağlayan yapılar da süreç içerisinde ulaşılması istenen amaçlardandır [104].

Anomali tespit aşaması, zeki fabrika sistemleri için önemli olan sorunlardan biridir. Zeki fabrikaların sahip olduğu karmaşık yapı, sistemlerin istenmeyen durumların oluşmasına sebep olabilmektedir. Son zamanlarda, özellikle güvenlik açıkları ve anomali konusu, zeki fabrikalar için fenomen halini almıştır [105]. Anomali tespiti için çalışma yapılan sistemlerde, bakım sıklığını en düşük seviyeye çekebilmek için tahmini bakım çalışmaları yapılabilir ve üretim kaynaklarının verimli kullanılması kapsamında uygulanan çalışmalarda önemli ölçüde maliyet avantajı sağlanabilir. Ayrıca fabrika içi üretim kapasitesi ve sistemin karmaşıklık seviyesi artıkça, karşılaşabilecek olan sorunların çeşidi ve miktarı da artış gösterecektir. Oluşması muhtemel bu sorunların çözümlerinde, sorunu erken tespit etmek ve maliyetleri en aza indirebilmek için, üretim sistemi içerisinde çalışan cihazların anormal davranışlarının analizlerinin yapılması ve tespit edilmesi gerekmektedir. Böylece üretim sisteminde yaşanacak süreç gecikmeleri ve zararların daha hızlı önüne geçmek mümkün olabilecektir [106].

3.3. Anomali Tespit Yaklaşımları

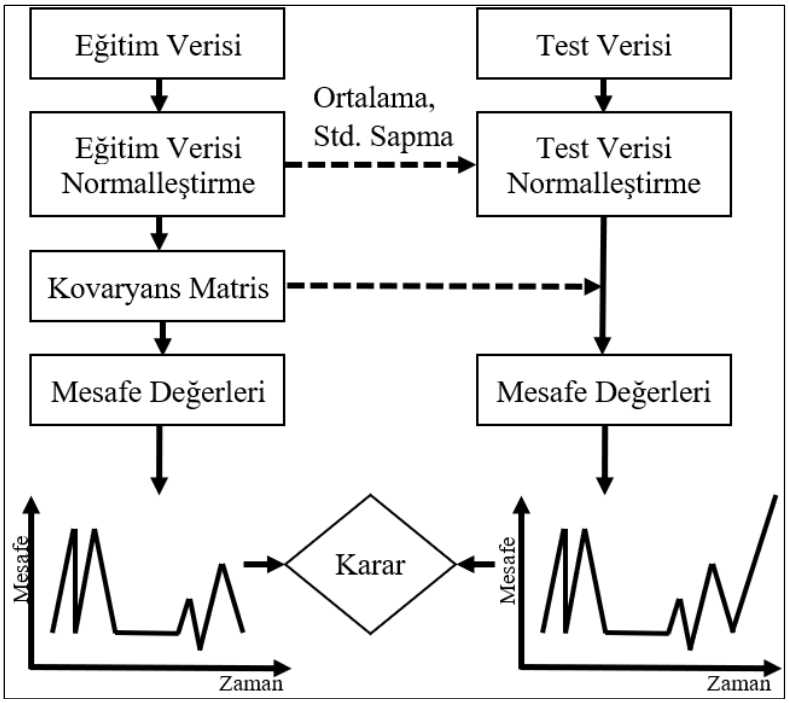
Bu başlık altında, anomali tespit sistemlerinde kullanılan yaklaşımların çeşitleri ve temel özellikleri açıklanmaya çalışılmıştır.

3.3.1. Mesafeye dayalı anomali tespit yaklaşımı

Bu yaklaşımda, veri seti içerisinde bulunan temsil verileri üzerinde, mesafeye dayalı olarak bir anomali puanı ataması yapılır. Bu işlemler için, ilk etapta yapılması gereken, tüm eğitim verilerini öğrenilen düşük boyutlu veriye dönüştürmek için daha önce eğitilmiş olan evrişimli otomatik kodlayıcının başlangıç modeli kullanılmaktadır. İkinci olarak da belirli bir test veri kümesi aynı uzaya dönüştürülmekte ve test verisi ile bu uzaydaki en yakın komşusu arasında mesafe ölçüm fonksiyonuna göre mesafesi hesaplanarak anomali puanı belirlenmektedir [107].

Mesafeye dayalı yaklaşımla anomali tespiti, bir veri kümesi içerisinde bulunan uzak örnekleri bulmaya çalışmaktadır. Test verileri, veri kümesinin tamamı içerisinde örnek olarak alınmakta ve test verilerinin örnekler tarafından oluşturulmuş olan alanda normal dışı bir değere sahip olup olmadığı kontrol edilmektedir [108].

Mesafeye dayalı yaklaşım, normal verilerden uzaktan bulunan örneklerin doğasını kullanmaktadır ve karar verme aşamasında bir mesafe ya da benzerlik ölçüsünün kullanımını gerektirmektedir. Mesafe tabanlı anomali tespiti genel olarak Şekil 3.1'de görüldüğü şekilde uygulanabilmektedir. Bu yöntemin ilk adımında, eğitim verilerin normalleştirilmesi için ortalama ve standart sapma değerleri hesaplanmaktadır. Bu aşamada z-skor veya farklı bir normalleştirme yöntemi uygulanabilir. Aynı şekilde test verileri için de normalleştirme uygulayabilmek için ortalama ve standart sapma değerleri hesaplanır. Sonrasında, normalize edilmiş eğitim verileri içerisinde mesafelerin hesaplanması için bir kovaryans matrisi elde edilmektedir. Oluşturulmuş olan kovaryans matrisi, normalleştirilmiş olan test verileri üzerinde uygulanarak mesafe hesaplamalarında kullanılmaktadır. Burada önemli olan kısımlardan biri de kontrol edilen test verilerinin anomali olup olmadıklarını belirleyebilmek için uygun bir eşik noktası belirlenmesidir [109].



Şekil 3.1. Mesafeye dayalı anomali tespit yaklaşımı [109]

Endüstri için gerçekleştirilen uygulamalarda, genellikle sadelikten yana bir tercih yapılmaktadır. Kullanılan yöntemler ne kadar karmaşık olursa, kullanıcılar tarafından kullanılan uygulamalar da o derece karmaşık bir sahip olacaktır. Bu durum da çoğunlukla bakım gereksinimlerinin artması için yeterli olabilir. Çoğunlukla basit mesafeye dayalı anomali tespit sistemleri, kullanılan değişkenlerin uygun şekilde ölçeklendirilmesi şartı ile yeterli performans sağlayabilir [110].

3.3.2. Kümeleme tabanlı anomali tespit yaklaşımı

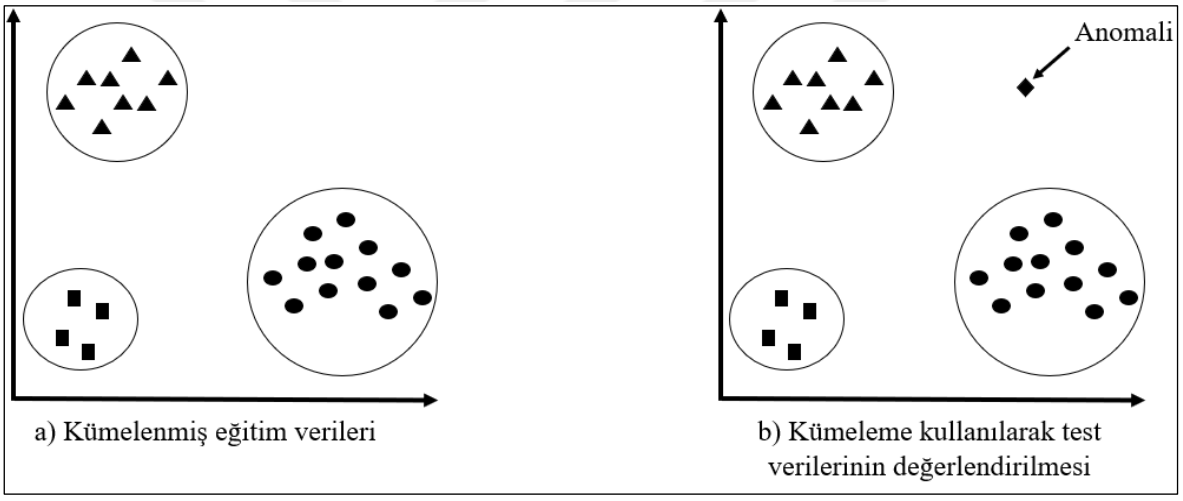
Kümeleme, denetimsiz bir öğrenme sürecidir ve benzer özelliklere sahip verileri bir sınır içerisinde bir araya getirilmesine dayanır. Verilerdeki benzerlik değerleri de aralarındaki mesafe ile ölçülmektedir [111]. Kümeleme, verilerin benzer nesneler grubuna bölünmesi ve dağıtılması olarak tanımlanabilir. Her küme, birbirine benzeyen ve diğer kümelerdeki verilere benzemeyen veri gruplarından oluşmaktadır. Kümeleme tabanlı anomali tespit sistemleri sayesinde, önceden bilinmeyen hataların tespit edilmesi gerçekleştirilebilir [7].

Kümeleme, çeşitli boyutlarda bulunabilen etiketlenmemiş veriler içerisinde örüntüler bulma eylemi olarak tanımlanabilir. Kümeleme tabanlı anomali tespit yaklaşımları için iki farklı yöntem bulunmaktadır. İlki, eğitim veri setinin hem normal hem de anormal verileri içerecek şekilde etiketlenmemiş verilerden oluşması ve bu veri seti ile denetimsiz eğitim süreçlerinin gerçekleştirilmesi şeklindedir. Bu yönteme, denetimsiz kümeleme ismi verilmektedir. İkinci yöntem ise eğitim sürecinin tamamen normal verilerden oluştuğu durumlarda gerçekleştirilen yarı denetimli kümeleme işlemidir. Denetimsiz kümeleme yönteminde, veri seti içerisinde bulunan anomali verileri, toplam verilerin küçük bir yüzdesini oluşturduğu varsayılmaktadır. Bu varsayıma göre göre, büyük boyutlu kümeler normal veriler tarafından oluşturulmakta, farklı noktalarda küçük boyutlu olarak ortaya çıkan kümeler ise anomali verilerini göstermektedir [112].

Kümeleme tabanlı anomali tespit yaklaşımı ile mesafeye dayalı anomali tespit yaklaşımı arasındaki temel fark, kümeleme tabanlı yöntemlerde her bir gözlemin ait olduğu kümeye göre değerlendirilmesi, mesafeye dayalı yöntemlerde ise her bir gözlemin yakın komşuluğuna göre analiz edilmesidir. Kümeleme tabanlı anomali tespit yaklaşımları, bir nesnenin değişen çalışma koşulları altında toplanan referans verilerinin kümelere ayrılması

ve her bir verinin referans kümeleri kullanılarak ele alınması şeklinde kullanılmaktadır [109].

Genel manada kümeleme tabanlı anomali tespit sistemlerinde kümelerin oluşturulması süreci iki ana aşamada gerçeklemektedir. İlk aşama veriler üzerinde K-Means, Fuzzy C-Means veya diğer kümeleme algoritmalarından birinin kullanılmasıdır. İkinci aşama da alt dizilerin farklı kümelere karşı olan uygunluklarının bazı ölçüm sonuçlarına göre belirlenerek bir anomali puanının belirlenmesidir. Anomali puanı, alt diziler ve küme merkezleri arasındaki mesafe değeri ile kolaylıkla atanabilmektedir. Bu düzende elde edilen verilerin normal veya anomali verisi olup olmadıklarının tespit edilmesi amaçlanmaktadır [113].



Şekil 3.2. Kümeleme tabanlı anomali tespit yaklaşımı örneği [114]

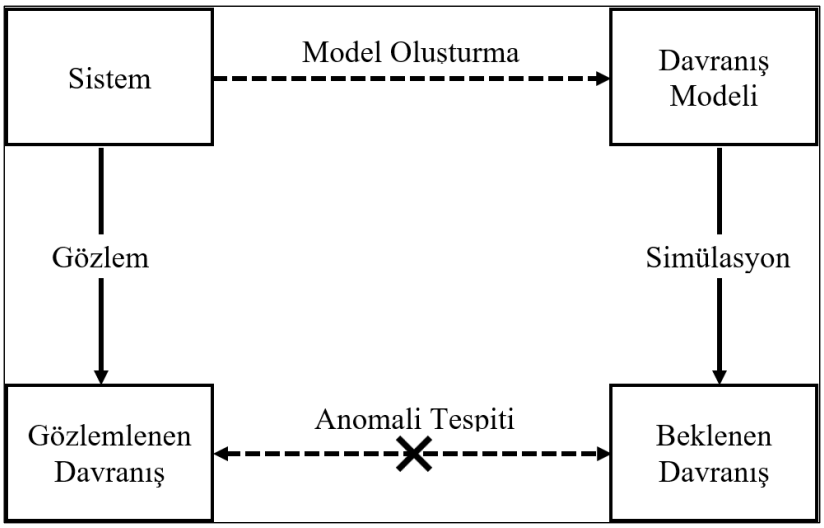
Bazı kümeleme algoritmaları, veri noktalarının tanımlanan kümelerin dışında kalmalarına müsaade etmektedir. Bu şekilde gerçekleştirilen eylemlerde herhangi bir kümeye ait olmadığı tespit edilen veriler, anomali olarak işaretlenmektedir (Şekil 3.2). Bölümleme şeklinde çalışan kümeleme algoritmalarında, noktaların tamamı uygun görülen bir kümeye yerleştirilir. Bu durumda her küme, boyut koşulunu karşılamaktadır. Ancak, kümelerin içlerinde bulunan bazı veri noktaları, aynı küme içerisinde bulunan diğer verilerden daha uzak noktalara konumlanabilirler. Bu şekilde tespit edilen veriler de anomali olarak değerlendirilmektedir [5].

3.3.3. Model tabanlı anomali tespit yaklaşımı

Birçok veri seti, ilgili değişkenler arasında varsayılan bir fonksiyonel veya ilişkisel bağlantıyı belirleyen, veri üretiminin gerçekleştirilmesini sağlayan modeller ile tanımlanmaktadır. Bu modeller sayesinde, veri kümeleri anlaşılabilir ve kısa bir özet şeklinde tanımlanabilmektedir. Bu sayede, normal veri düzenleri ile tutarlı olmayan verilerin tespit edilmesi ve ayrıştırılması sağlanabilmektedir. Örneğin D veri seti, bir MD modeli tarafından tanımlandığı zaman, anomali tespit işlemleri ya model parametreleri tarafından ya da veri alanı tarafından gerçekleştirilmelidir. İlk seçenekte odak, öğrenilmiş olan model parametreleri ve bu parametrelerin bir veri noktasından ne derecede etkilendiği bilgisidir. İkinci seçenekte ise farklı veri noktaları arasındaki göreceli anormallikler devreye girmektedir. Anomali tespiti için bu veri noktaları ve model tahminleri arasındaki varyasyonun büyüklükleri karşılaştırılmaktadır. Model tabanlı anomali tespit sistemlerinde, modelin oluşturulması aşamasında aşağıdaki soruların cevaplanması gerekmektedir [5]:

- Model nasıl temsil edilecek?
- Bir veri noktasının modele ne kadar uyduğu nasıl değerlendirilecek?
- Bir anomali oluşması için şartlar neler?
- Mevcut verilerden model parametreleri nasıl üretilir veya nasıl öğrenilir?

Model tabanlı anomali tespit sistemleri, yapay zekanın önemli uygulamalarından biridir. Bir sistemin tanımlanması ve davranışlarının gözlemlenmesi ile sistemin anormal davranışlar gösterip göstermediğini tespit edebilmek için algoritmik analizler gerçekleştirmektedir. Şekil 3.3, model tabanlı anomali tespit sistemlerinin temel yapısı göstermektedir. Model tabanlı anomali tespit sistemlerinde, örnek olarak bir üretim sisteminin simülasyonu oluşturulur. Sonrasında simülasyon tarafından tahmin edilerek oluşturulan sistem davranışı ile gerçekte uygulanan sistemin davranışları karşılaştırılarak anomali tespit süreci uygulanmaya çalışılır. Model tabanlı anomali tespit teknikleri, üretim sistemleri içerisinde oluşabilecek enerji anomalileri, web servis süreçlerinde hata analizleri, elektrikli güç sistemlerinde ve otomotiv sistemlerinde anomali tespiti olarak kullanılmaktadır [115].



Şekil 3.3. Model tabanlı anomali tespit sisteminin temel yapısı [115]

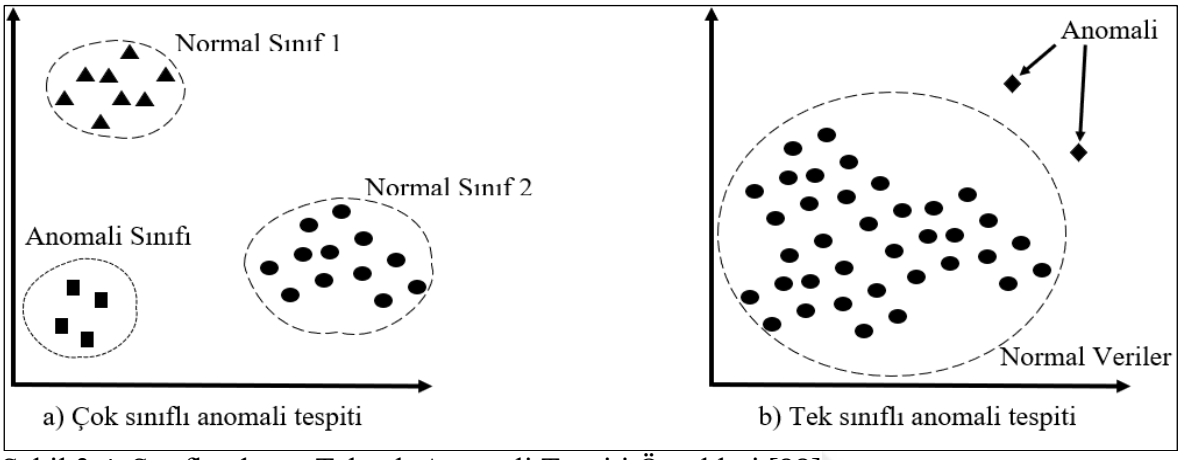
Model tabanlı anomali tespit sistemlerinde, anormal durumların analizi, izlenen sistem hareketleri ile oluşturulan performans temel modeli olarak adlandırılan simülasyon tarafından tahmin edilen hareketlerin karşılaştırılması şeklinde gerçekleştirilmektedir. Sistemde herhangi bir anormal durum söz konusu değilse, sistem hareketleri ile simülasyon tahminlerinin birbirlerine çok benzer şekilde gerçekleşmeleri gerekmektedir. Bu tür durumlarda atık veriler çok küçük seviyede kalabilirler. Ancak sistemde anomali oluştuğu zaman, sistem hareketleri ile simülasyon tahminleri arasında önemli derecede farklar ortaya çıkacak ve atık verilerin oranı da aynı şekilde artacaktır [116].

3.3.4. Sınıflandırma tabanlı anomali tespit yaklaşımı

Sınıflandırma, bir dizi etiketli veri örneğinden bir öğrenme modeli ortaya çıkartılması ve sonrasında test verileri içerisinde bulunan her veriyi söz konusu sınıflardan birine sınıflamak için kullanılmaktadır. Sınıflandırma tabanlı anomali tespit sistemleri de benzer yapıda çalışmaktadır. Normal ve anomali olarak etiketlenmiş eğitim verileri ile model oluşturulur. Test aşamasına geçildiğinde veriler okunarak hangisinin normal olduğu, hangisinin de anomali olduğu konusunda sınıflama işlemleri gerçekleştirilir. Sınıflandırma tabanlı anomali tespit yaklaşımları, mevcut etiketleme yapısına dayalı olarak iki kategoride gerçekleştirilebilir. Bunlar [109]:

- Tek sınıf sınıflandırma**,** sınıflandırma tabanlı anomali tespit yaklaşımları için tek sınıf olarak belirtilmiş olan anomali etiketlerini, normal veriler etrafında ayrımcı bir sınır öğrenmek için kullanmaktadır.

- Çok sınıflı sınıflandırma, sınıflandırma tabanlı anomali tespit yaklaşımları için her bir normal sınıf ile diğer sınıflar arasında ayrım yapabilmek için kullanılmaktadır. Bu yaklaşımda okunan bir veri, herhangi bir sınıflayıcı tarafından normal olarak atanmamışsa, anomali olarak belirlenmektedir.



Şekil 3.4. Sınıflandırma Tabanlı Anomali Tespiti Örnekleri [98]

Sınıflandırma, sınıf etiketi bilinen örnekler içeren bir eğitim setinin temelinde gerçekleştirilen ve yeni karşılaşılan verilerin sınıf etiketini belirleme süreci olarak tanımlanabilmektedir. Sınıflandırmada kullanılan çeşitli veriler, bir veya daha fazla sınıf etiketine sahip olabilirler. Sınıflandırma, makine öğrenmesi kapsamında denetimli öğrenmenin bir parçası olarak kabul edilmektedir. Örnek olarak doğru tanımlanmış verilerden oluşan bir eğitim setinin kullanılması gösterilebilir. Sınıflandırma uygulayan algoritmalara da sınıflandırıcı adı verilmektedir. Anomali tespit sistemlerinde genellikle, normal veriler ve anomali verileri olmak üzere, eğitim setinde etiketler oluşturularak sınıflandırma tabanlı anomali tespit sistemleri kullanılmaktadır [7].

Anomali verileri, normal verilerden farklı olmalıdır. Bu tür problemler, sınıflandırma kapsamında genellikle tek sınıflı sınıflandırma yöntemi ile değerlendirilmektedir. Örnek olarak sistemin normal davranış düzeni modellenir, eğitim modeli ortaya çıkartılır ve sonrasında karşılaşılan yeni verilerin normal verilerden mi, yoksa anomali mi olup olmadığı tespit edilmeye çalışılır. Bu yapıda, sistemin yeni karşılaşacağı verileri normal veya anomali olarak sınıflandırabilmesi için bir karar sınırı sağlayacak olan uygun bir eşik noktasının belirlenmesi gerekmektedir [117].

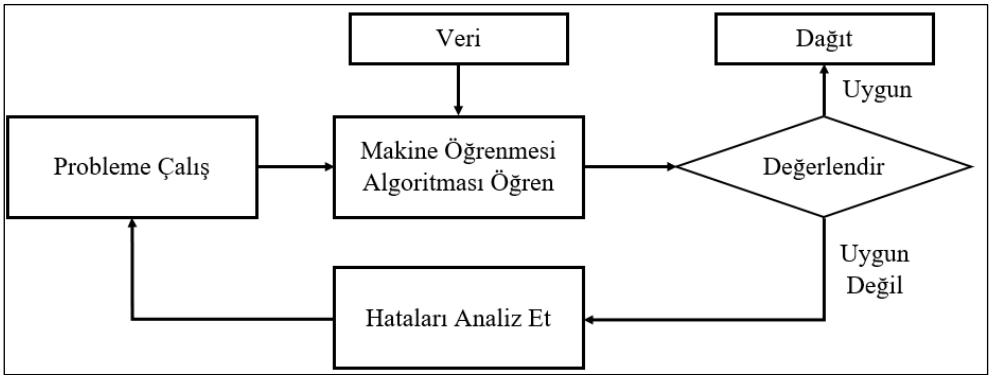
Sınıflandırma tabanlı anomali tespit yaklaşımları, verilerde izlenen normal davranışların, anormal davranışlardan ayırt edilebileceği fikrine dayanmaktadır. Normal ve anormal davranışları birbirlerinden ayırt edebilmek için de bir sınıflandırıcı kullanılması gerekmektedir. Sınıflandırıcı tarafından yeni elde edilen veriler üzerinde incelemeler yapıldığında, yapılan tahmin normal verilerin gösterdiği davranışa uymadığı takdirde, anomali olarak işaretlemesi yapılmaktadır. Sınıflandırma işlemi, genellikle dört aşamadan oluşmaktadır. Birinci aşama, eğitim verilerinde mevcut sınıfların belirlenmesi ve verilerin bu sınıflara göre etiketlenmesidir. İkinci aşamada, sınıflandırmanın gerçekleştirilebilmesi için verilerin öznitelikleri tanımlanmaktadır. Üçüncü aşamaya geçildiğinde, hazır duruma getirilen eğitim verileri kullanılarak anomali tespiti gerçekleştirilmesi planlanan öğrenme modeli oluşturulmaktadır. Son aşamada ise oluşturulan öğrenme modeli kullanılarak, karşılaşılan yeni verilerin normal veya anomali verisi olup olmadıkları tespit edilmeye çalışılmaktadır [99].

Sınıflandırma tabanlı anomali tespit sistemlerinde çok değişkenli veri örnekleri ve alt dizileri iki sınıfa ayrılmaktadır. Bu sınıflar normal veriler ve anomali verileri olarak belirtilir. Sınıflandırıcı, veri setinden ayrılan bir bölümü kullanarak eğitim süreçlerini gerçekleştirir ve test verileri üzerinde kontrol ettiği veri özelliklerini değerlendirerek her veri için anomali puanını belirler. Bu yaklaşımı kullanmanın avantajı, çoğunlukla anomali tespitindeki doğruluk oranları, denetimsiz yaklaşımlara göre nispeten daha yüksek olmaktadır. Ancak sınıflama tabanlı anomali tespit sistemi uygulayabilmek için kullanılacak olan eğitim verilerinin toparlanması zaman alıcı bir süreç olarak gerçekleşmektedir [113].

4. MAKİNE ÖĞRENMESİ

Gelişmiş ve hızlı teknoloji sayesinde, birçok kaynakta bilgi çağı olarak da adlandırılan bir zamanda bulunmaktayız. Her geçen gün makine öğrenmesi, yapay zeka gibi kavramlarla ilgili yeni araştırma çalışmaları yapılmakta ve sonuçlar tartışılmaktadır. Özellikle ucuz ve büyük miktarda bilgi işlem, yüksek kapasiteli donanımlar ve depolama birimleri sayesinde bu alanlar etrafında büyüyen bir araştırma ekosistemi ortaya çıkmıştır. Araştırmacılar ve veri bilimciler tarafından, olayları tahmin edebilen, karmaşık analizleri hızlı ve doğru bir şekilde gerçekleştirebilen, anomali tespiti yapabilen zeki modeller ve sistemler oluşturmak için daha verimli yeni teknikler geliştirmek için çalışmalar devam etmektedir [118].

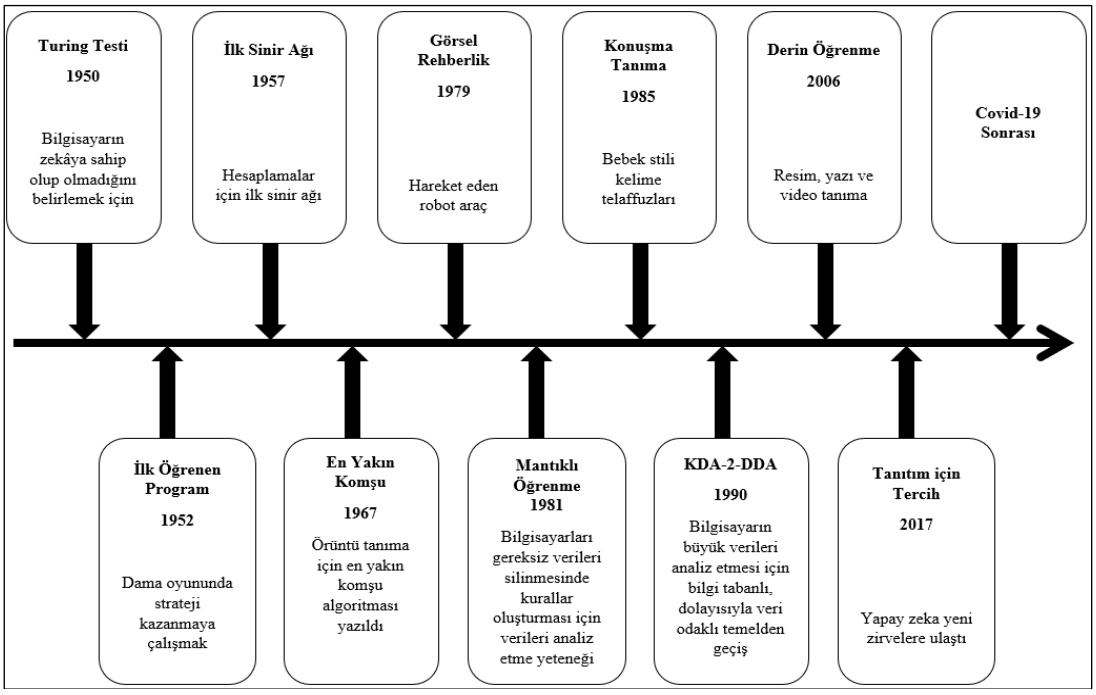
Çeşitli sistemlerden ve sensörlerden son zamanlarda elde edilen büyük miktarlardaki veriden, iç görü elde edebilmek için makine öğrenmesi teknikleri kullanılabilmektedir. Veriye dayalı olarak anomali tespit işlemleri de makine öğrenmesi kullanılarak yapılabilir. Makine öğrenmesi, örnek olarak bir tanılama sistemi, hata örneklerini içerisinde barındıran bir eğitim veri setinden ilgili sürecin hatalarını belirlemeyi öğrenen bir yazılım sistemidir. Öğrenme modellerinin oluşturulması için kullanılan veri gruplarına eğitim kümesi, oluşturulan öğrenme modellerinin test edilebilmesi için kullanılan veri gruplarına da test kümesi adı verilmektedir. Makine öğrenmesi tekniklerine dayalı olarak geliştirilen bir anomali tespit sistemi, eğitim veri kümesi içerisindeki hata türlerini tespit ederek, hangi özelliklerin veriler içerisindeki hataları daha iyi tespit edebildiğini iyi tahmin edici araçları otomatik olarak öğrenmektedir. Şekil 4.1'de anomali tespitine yönelik makine öğrenmesi yaklaşımı için üst düzey bir genel bakış gösterilmektedir [9].



Şekil 4.1. Anomali tespitine yönelik makine öğrenmesi yaklaşımı için üst düzey bir bakış [9]

Makine öğrenmesi, insanlar tarafından gerçekleştirilemeyecek kadar büyük ve karmaşık yapıdaki görevlerin uygulanabilmesine imkan tanımaktadır. Öğrenme kavramı, bir görevin başarı ile yerine getirilmesi ile ilgilenmektedir. Makine öğrenmesi için belirlenen görevler ise, bir uygulamanın örneğinin nasıl işlendiği ile ilgili olmaktadır. Makine öğrenmesi algoritmaları çok miktarda bulunan öğrenme kümesi için belirlenmiş verileri analiz eder. Küme içerisindeki verilerin tamamının özelliklerini karşılaştırır ve öğrenme sürecini gerçekleştirir [8].

Bilgi teknolojileri gelişmesinden dolayı artan bilgi miktarı ile daha kesin ve eksiksiz öğrenme ve anlayışların yaratılabilmesi için çok büyük miktarlardaki verilerin düzenlenmesi ve incelenmesine karşı bir odak oluşmuştur. Gelecek dönemlerde bilgi hacimlerinin ve bilgilerin buluta aktarımlarının seviyesi daha da artacaktır. Bilgi seviyeleri artıkça daha fazla işletme tarafından işlenecek, analiz edilecek ve fayda sağlanmaya çalışılacaktır. Devasa miktarlarda ortaya çıkan verilerin depolama ve inceleme şansı elde edildikçe çok daha fazla faydalı bilgiye erişim mümkün olacaktır. Uzmanlar, bulut sistemlerinin ve bilgi edinmenin çeşitli süreçlerinde kullanılan bilgisayarların kapasitelerinin, daha fazla gelişmiş yardımsız hesaplamalar, daha kapsamlı kişiselleştirme ve öznel yönetimler nedeniyle büyük ölçüde artacağını ifade etmektedirler. Şekil 4.2'de makine öğrenimi kavramının yıl bazında gelişimi gösterilmektedir [119].



Şekil 4.2. Makine öğrenmesi tekniklerinin evrimi [119]

Makine öğrenmesi, toplu halde elde edilen veri setleri kullanılarak gerçek dünyadaki sorunları çözebilmek ve modeller oluşturmak için kullanım imkanı olan birden fazla algoritma tekniğine sahiptir [118]. Öğrenme sürecinde insan denetiminin oranına dayalı makine öğrenmesi yöntemleri şunlardır:

- Denetimli öğrenme**,** tahmin işlemlerini gerçekleştirmek için kullanılmaktadır. Bu öğrenme türünde amaç, belirli bir ilgi sonucunu tahmin etmek ve sınıflandırma yapmaktır [120]. Sınıflandırma, ön bilgilerden gelen veri setlerini sınıflara ayırdığından dolayı denetimli bir öğrenme yöntemidir. Her test verisinde, özelliklerin birleştirilmesinde ve eğitim verilerinden her sınıf için ortak örüntülerin ortaya çıkartılması ile sınıflar konusunda karar verilmektedir. Sınıflandırma işlemi iki aşamada gerçekleştirilmektedir. İlk olarak eğitim veri setine sınıflandırma algoritması uygulanır ve öğrenme modeli ortaya çıkartılır. Sonrasında ise model test veri seti üzerinde denenerek doğrulanması gerçekleştirilir [121].
- Denetimsiz öğrenme**,** ölçülen bir sonuca ait olmayan bir veri yapısındaki ilişkileri bulmayı amaçladıklarından dolayı açıklama görevlerinde önemli bir yere sahiptir. Bu yöntemde analiz sonucunu kontrol edebilecek bir değişken olmadığından dolayı denetimsiz olarak adlandırılmaktadır. Amaç, bir veri seti içerisindeki boyutları, bileşenler ve kümeleri ortaya çıkartmaktır [120].
- Yarı denetimli öğrenme**,** hem etiketlenmiş hem de etiketlenmemiş olan verilerden oluşan veri setleri üzerinde kullanılan bir makine öğrenmesi tekniğidir. Etiketlenmiş olan veriler, genel olarak veri seti içerisindeki küçük bir oranı temsil etmektedir. Bu öğrenme tekniğinin amacı, etiketlenmiş ve etiketlenmemiş verilerin bir arada analiz edildiğinde öğrenme davranışını nasıl etkileyeceğini tespit edebilmek ve bu tür kombinasyonlar kullanan algoritmalar geliştirebilmektir [122].
- Pekiştirmeli öğrenme**,** bir hipotez tarafından oluşturulan tahminlerin, gelecek uygulamalar oluşturulmasını etkilediği yapıları incelemektedir. Bu tür uygulamalar, programlanabilir bir sistemin farklı zamanlardaki durumlarını tespit etmeye çalışan veri noktalarını içermektedir. Bu veri noktalarında oluşturulan etiketler, sistemin belirli bir durumda gerçekleştirmesi gereken en uygun eylemi göstermektedir. Çalışma yapısı olarak bir miktar denetimsiz öğrenme tekniği ile benzerlik göstermektedir. Pekiştirmeli öğrenmenin de etiketli verileri kullanmadan öğrenme sürecini tamamlaması gerekmektedir [123].

Kabaca ifade etmek gerekirse makine öğrenmesi, analiz edilen veri setlerinin özelliklerine bağlı olarak, yeni karşılaştığı verilerin etiketleri ile ilgili tahminlerde bulunmayı amaçlamaktadır [123]. Günümüz teknolojisinde makine öğrenmesi algoritmaları, karmaşık yapıdaki problemlerin çözülebilmesi noktasında büyük umut vaat etmektedir. İnternet aramalarından ses tanıma sistemlerine kadar günlük kullanıma açık olan uygulamalardan bazıları, makine öğrenmesi algoritmalarının vaatlerini gerçekleştirme noktasında atılan önemli adımların örnekleri olarak karşımıza çıkmaktadır [124].

4.1. Makine Öğrenmesi Algoritmaları

Tez kapsamında veri setleri üzerinden yapılan analiz çalışmalarında, Naive Bayes, Yapay Sinir Ağları, Lojistik Regresyon, Destek Vektör Makineleri, Karar Ağaçları, Rastgele Orman ve k En Yakın Komşu algoritmaları ile öğrenme modelleri oluşturulmuştur. Ayrıca AutoML kütüphaneleri ile yapılan testlerde Gradient Boosting Machine ve Light Gradient Boosting Machine algoritmalarının da sıklıkla öğrenme modelleri oluşturulması için kullanıldığı gözlemlenmiştir. Aşağıdaki başlıklarda bu algoritmalar ile ilgili açıklamalar yapılmıştır.

4.1.1. Naive bayes (NB)

İstatistiksel bir sınıflandırma algoritması olan Naive Bayes (NB), arka planda istatistiksel değerlere göre farklılık gösterebilen bir çalışma sistemine sahiptir. Bu sebeplerden dolayı dinamik olarak çalışan sistemler üzerinde kullanımı esnasında, tekrar tekrar hesaplama işlemlerinin gerçekleştirilmesini gerektiren bir algoritmadır [125].

Naive Bayes sınıflandırma tekniği, Bayes kuralı ile birlikte karar ağaçları modeli birleştirilerek elde edilmiş bir tekniktir. Naive Bayes Algoritması, örneği verilmiş olan her sınıfın olasılık değerini hesaplamak amacıyla Bayes kuralını kullanmaktadır [126]. Makine öğrenmesi uygulamalarında sıkça karşılaşılan bir sınıflandırma tekniği olan Naive Bayes, koşullu olasılık hesaplamaları üzerine çalışan ve Bayes kuralının en basit hali olarak nitelendirilen bir algoritmadır [127]. Naive Bayes, sınıflandırma teknikleri içerisinde en kısıtlayıcı alanda yer almaktadır. Bu algoritmada örnek verilerin hangi sınıfa ait oldukları bilinmektedir. Genel anlamda metin sınıflandırılmasında üstün başarı gösterdiği tespit

edilmiştir [128]. Naive Bayes algoritması, koşullu sınıf bağımsızlığının varsayımı üzerinde durmaktadır. Yani, herhangi bir değer niteliğinin, diğer niteliklerin değerlerinden bağımsız olduğu düşünülür. Pratikte nitelikler arasında bir miktar bağımlılık olsa dahi, hesaplamaların kolaylaştırılması amacıyla teoride bu varsayım uygulanmaktadır. Eğer uygulanan varsayımlar doğru olursa, Naive Bayes algoritması diğerlerine göre en iyi sonucu veren algoritma olacaktır. Ayrıca işleme tabi tutulan nitelikler arasında öneme göre bir derecelendirme yapılmaz ve sınıfın tahmin edilmesinde tüm niteliklerin aynı derecede önemli olduğu kabul edilir [129].

Genel anlamda Naive Bayes tekniği X=(X1, X2,…,Xn) örneğindeki verinin C sınıfına ait olup olmama olasılığını hesaplamak için tercih edilmektedir [130].

$$P(h1|xi) = \frac{P(xi|h1)P(h1)}{P(xi|h1)P(h1)+P(xi|h2)P(h2)} \quad (4.1)$$

Eş. 4.1'de, P(h1) ifadesi, h1 hipotezi ile birlikte ön olasılık olduğu zaman, p(h1 | xi) ifadesi sonraki olasılık olarak değerlendirilmektedir [131].

Ayrıca Eş. 4.2 ile P(h1|xi) değeri maksimize edilmektedir. Maksimize edilmiş olan P(h1|xi) değerinin C sınıfı, Bayes teoremine göre maksimum sonraki olasılık olarak ifade edilmektedir [129].

$$P(h1|xi) = \frac{P(xi|h1)P(h1)}{P(xi)} \quad (4.2)$$

4.1.2. Yapay sinir ağları (YSA)

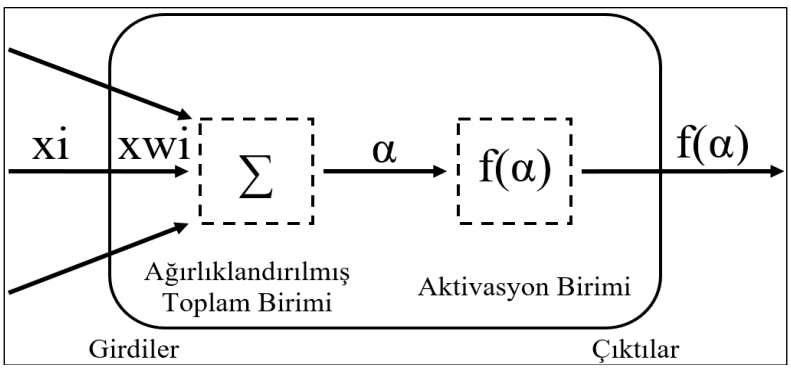
Yapay Sinir Ağları, ağırlıklı bağlantıları kullanarak birbiri ile bağlantı kurmuş olan elemanlardan oluşmaktadır. Ayrıca bu elemanların, her biri kendilerine ait, paralel ve dağıtılmış bilgi işleme yeteneğine sahip bellekleri bulunmaktadır. Farklı bir ifade ile Yapay Sinir Ağları, biyolojik sinir sisteminin kopyası gibi çalışması amaçlanmış bir bilgisayar programıdır. Yapay Sinir Ağları, bu özelliği sayesinde kendi kendine öğrenen bir yapıya sahiptir. Öğrenmenin yanında ezberleme, bilgiler arasındaki ilişkiyi ortaya çıkartma gibi

yetenekleri de bulunmaktadır. Ayrıca tüm bunları yapabilmesi konusunda yazılımcının geleneksel yeteneklerine muhtaç değildir [46].

YSA, yapay zeka çalışmalarının gelişmesine katkı sağlayan alanlardan bir tanesidir. Buna dayanarak YSA'nın, öğrenme yeteneğine sahip sistemlerin başında gelen yapay zeka teknolojilerinin bir parçası olduğu düşünülebilir. YSA için, insan beyninin temel elemanlarından olan nöronların çalışma prensiplerini kopyalamaya çalışarak, gerçek sinir sistemi yapısının bir simülasyonunu oluşturmaya yarayan programlar olduğu söylenebilir [132].

Yapay Sinir Ağları, insan beyninde bulunan nöronlar gibi çalışan yapay nöronlar aracılığı ile örnekler üzerinde yeterli incelemeleri yaparak değerli olan bilgiyi ortaya çıkartmak için kullanılmaktadır. Bu yapay nöronlar, problemlerin çözüm aşamalarında kendi öğrendiklerini kullanarak karar verebilme yeteneğine sahiptirler. Kısaca Yapay Sinir Ağları, çeşitli geometrik şekillere sahip yapay nöronların arasında kurulan bağlantı ile oluşan ağ yapıları olarak ifade edilebilmektedir. Bahsi geçen ağ yapıları oluşturulduktan sonra, yeni gelen verilerin sınıflandırılması için kullanılmaktadırlar [133]. Yapay Sinir Ağları sıklıkla, doğrusal olsun veya olmasın herhangi bir problem hakkında girdi olarak kullanılan veriler ile çıktı olarak elde edilmesi gereken veriler arasında gerekli bağlantıyı kurarak sonraki uygulamalar hakkında sonuçlar elde edebilmek amacıyla kullanılmaktadır [134].

Basit problemlerin çözümünde kullanılabilecek olan sinir ağı, girdi ve çıktı nöronlarından oluşan tek katmanlı ağlar olarak ifade edilmektedir (Şekil 4.3).

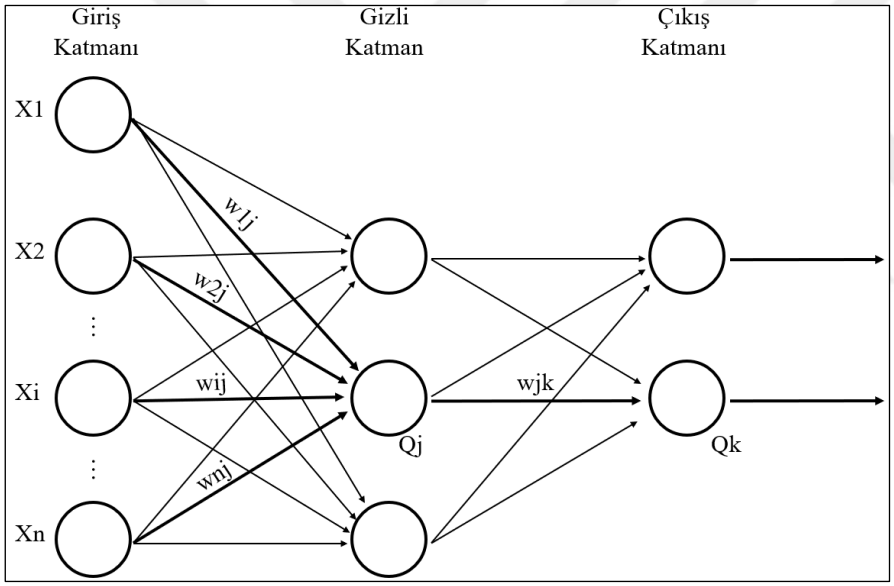


Şekil 4.3. Tek katmanlı yapay sinir ağı yapısı [134]

Bu tip ağlarda tüm girdi ve çıktı nöronları bir veya birden fazla olabilir ve tüm girdi nöronları, tüm çıktı nöronlarına ağırlıklandırılarak bağlanmaktadır. Ağırlıklandırılmış bağlantı, eşik değeri ile kontrol edilerek aktivasyon fonksiyonun çalıştırılması ve çıkış değerinin hesaplanması hedeflenmektedir. Bu olay matematiksel olarak Eş. 4.3'te ifade edilmektedir [134].

$$f(\alpha) = \sum_{i=1}^{m} wixi + \theta \tag{4.3}$$

Karmaşık problem çözümlerinde ise tek katmanlı ağlar yeterli olamayacağından dolayı girdi ve çıktı katmanlarının arasında gizli katmanların devreye girdiği çok katmanlı ağlar kullanılmaktadır (Şekil 4.4).



Şekil 4.4. Çok katmanlı yapay sinir ağı yapısı [129]

Bu yapıda ağ, bir veri grubunun sınıf etiketini tahmin edebilmek için katmanlar arasında görev yapan ağırlıkların gerçek değerlerini öğrenme yoluna gider. Giriş katmanında bulunan nöronlardan gelen giriş verileri ağırlıklandırılarak gizli katmanda bulunan nöronların değerleri hesaplanır. Sonrasında ise gizli katmanda bulunan değerler yine ağırlıklandırılarak çıkış katmanı verisinin hesaplanmasına çalışılır [129]. Öğrenme modelinin oluşturulması aşamasında, elde edilen çıktı katmanı verisi ile beklenen çıktı katmanı verisi karşılaştırılarak hata miktarı hesaplanır. Ortaya çıkan hata miktarı doğrultusunda geri besleme yapılarak hata katsayıları hesaplanmaktadır ve sonrasında elde edilen katsayılar ile nöron bağlantılarında kullanılan ağırlık miktarları değiştirilmektedir.

Sistem, hesaplanan çıktı katmanı verisi ile beklenen çıktı katmanı verisi arasındaki hata miktarı en aza indirilene kadar bu işlemleri tekrarlamaya devam eder [135].

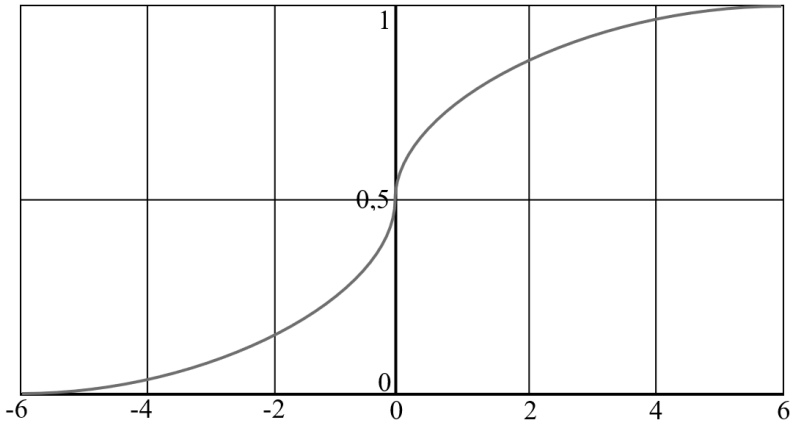
4.1.3. Lojistik regresyon (LR)

Lojistik Regresyon (LR) veri seti içerisinde bulunan tüm değişkenleri sayısal kabul eden ve normal bir dağılıma sahip olan, ikili sınıflandırma algoritması olarak tanımlanmaktadır. Bu varsayımda belirtilen son noktaya rağmen, LR ile normal dağılım göstermeyen veriler üzerinde dahi iyi sonuçlar alınabilmektedir. LR algoritması, doğrusal olarak bir regresyon fonksiyonun içerisinde birleştirilmiş ve bir lojistik fonksiyon kullanılarak dönüştürülmüş her giriş değeri için bir katsayı öğrenmeye dayalı bir sistem üzerine kurulmuştur. Hızlı ve basit bir sisteme sahip olmasına rağmen, bazı problemler üzerinde son derece etkili sonuçlar vermektedir. LR sadece ikili sınıflandırma modellerini desteklemektedir [136].

LR, düzeltilmiş olasılık oranları hakkında çıkarım sağlamak için geliştirilmiş standart bir öğrenme algoritmasıdır [137]. LR, Veri setinin kalitesine bağımlı olarak ayrıcı özelliğine sahip bir modeldir. Modelin belirlenmesinde, özellik değerleri (X1, X2,…,Xn), ağırlık değerleri (W1, W2,…,Wn), sapma değerleri (b1, b2,…,bn) ve sınıflar (1 / 0) dikkate alındığında Eş. 4.4 kullanılabilmektedir [105]:

$$Tahmin\ Edilen\ Değer = p(y = C|X; W, b) = \frac{1}{1+exp\ (-w^{transpose}\ X-b)} \quad (4.4)$$

LR, logaritmik ilerleme gerçekleştiren, logit bir fonksiyondur. LR, yapı itibari ile Şekil 4.5'teki grafikte gösterilen eğriye en iyi şekilde benzeyecek olan değerleri tespit etmeye çalışmaktadır [125].



Şekil 4.5. Lojistik regresyon grafiği [125]

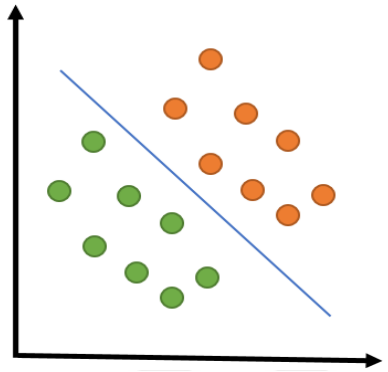
4.1.4. Destek vektör makineleri (DVM)

Destek Vektör Makineleri algoritması, LR algoritması gibi ayrımcı bir model olarak çalışmaktadır. Regresyon, aykırı veri tespiti ve sınıflandırma işlemleri için denetimli bir öğrenme sistemi sunmaktadır [105]. DVM, 1995 yılında Vladimir Vapnik tarafından tanıtılan, sınıflandırma ve örüntü tanıma süreçleri için kullanılabilen, basit ve verimli bir algoritmadır. DVM çalışma sisteminin ana amacı, hiper düzlemleri ve sınırları ortaya çıkartacak fonksiyonların elde edilmesidir. Oluşturulan hiper düzlemler, istatistiksel öğrenmeyi kullanmak amacı ile belirli algoritmalar ile eğitilerek, giriş verisi noktalarının farklı kategorilere ayrıştırılması sağlamak için kullanılmaktadır [138].

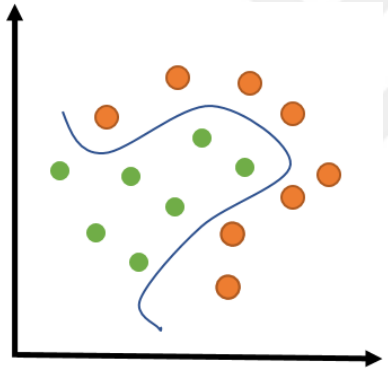
Destek Vektör Makineleri, prensip olarak istatistiksel öğrenme yöntemleri ve yapısal risklerin minimize edilmesine dayalı bir öğrenme algoritmasıdır. Hızlı öğrenme kapasitesine sahip büyük giriş verilerini kullanabilmesi, algoritmanın temel avantajını ortaya çıkartmaktadır. Bu sayede DVM algoritması ile doğrusal olmayan yüksek boyutlu veri modelleme sorunlarına çözüm getirilebilmektedir [139].

DVM, çoğunlukla sınıflandırma işlemlerinde kullanılan bir algoritmadır. DVM algoritmasının çalışma sisteminde, n bağımsız değişken sayısı olmak üzere tüm veri noktalarının n-boyutlu uzaydaki koordinatları değişken değerleri olarak belirlenmektedir. Bir üst aşamaya geçildiğinde ise iki sınıfı birbirinden ayıran iki boyutlu hiper düzlem tespit edilir ve sınıflandırma işlemi gerçekleştirilir (Şekil 4.6). Fakat, elde edilen tüm verilerin doğrusal hiper düzlemler ile sınıflandırılması mümkün değildir. Doğrusal olmayan veri türleri, DVM sınıflandırılmasında kernel fonksiyonu kullanılarak yüksek boyutlu uzayda

oluşturulan doğrusal olmayan bir hiper düzlem ile birbirlerinden ayrılabilir duruma getirilebilmektedirler (Şekil 4.7). Bu konumda uygulanan kernel fonksiyonu, doğrusal modda çalışan bir sınıflandırma algoritmasının, doğrusal olmayan bir problemi çözmesini sağlayacak olan kernel hilesi anlamında kullanılmaktadır [8].



Şekil 4.6. Doğrusal hiper düzlem ile sınıflandırma [8]



Şekil 4.7. Doğrusal olmayan hiper düzlem ile sınıflandırma [8]

4.1.5. Karar ağacı (KA)

Karar Ağaçları (KA) sınıflandırma ve regresyon problemlerinin çözümlerini destekleyebilmektedir. Veri örneklerini değerlendirmek amacıyla bir ağaç yapısının oluşturulması sistemine dayanmaktadır. Ters çevrilmiş bir ağacın kökü ile yapı başlar ve bir tahmin sonucuna ulaşılana kadar aşağıya doğru devam eder. KA, oluşturulması aşamasında, doğru tahminlere ulaşılabilmesi amacıyla en iyi ayırıcılık özelliğine sahip olan niteliğin tespit edilmesi süreçleri gerçekleştirilmektedir [136].

Dağılım tabanlı tahminleme işlemlerinde, giriş verilerinin tamamı üzerinde bir modeli geçerli olduğu varsayılır ve veri setine ait parametrelerin öğrenilebilmesi için de bütün veri seti kullanılır. Öğrenme işleminin sonrasında, test verilerinde sistemin sınanması

aşamasında da aynı yapının ve öğrenilmiş olan parametrelerin kullanımına devam edilir. Dağılıma bağımlı olmayan tahminleme süreçlerinde ise belirlenmiş bir ölçüt (ör: Öklid) ile öğrenme seti yerel parçalara ayrılmakta ve giriş verisi için, kendi alanına denk gelen verilerle eğitilmiş lokal bir model kullanılmaktadır. KA, yapıları gereği dağılımdan bağımsız çalışmaktadır. Çünkü öğrenme süreçlerinin başında, sınıf dağılımları ile ilgili tahminlerde bulunmaz. Karar ağaçlarında, ağacın yapısı baştan belirli olmamaktadır. Veri setinin özelliklerine ve yapısına göre dallar ve yapraklar eklenerek oluşturulmaktadır [140].

Karar ağacı oluşturulması konusunda, C4.5 ve ID3 gibi çeşitli algoritmalar kullanılmaktadır. Bu çalışma kapsamında C4.5 algoritması kullanılmıştır. C4.5, giriş verilerinin sıklıklarına göre sınıflandırma işlemini gerçekleştiren bir algoritmadır. Entropi hesabına dayalı olarak çalışmaktadır. C4.5 çalışma yapısı şu şekilde özetlenebilir [141]:

- Eş. 4.5 kullanılarak veri setinin sahip olduğu tüm nitelikler üzerinde entropi hesabı yapılır,
- Her bir nitelik için hesaplanmış olan entropi değeri, sınıfın entropi değerine bölünerek, niteliklerin bilgi kazanç değerleri ortaya çıkartılır,
- En yüksek bilgi kazancı değerine sahip olan nitelik kök olarak seçilir ve dağılım başlar,
- Kök olarak seçilen niteliğin sahip olduğu değerler haricinde kalan diğer nitelikler için aynı işlemler tekrarlanır,
- Tüm veriler işlenip yapraklara ulaşılınca ağaç tamamlanmış olur.

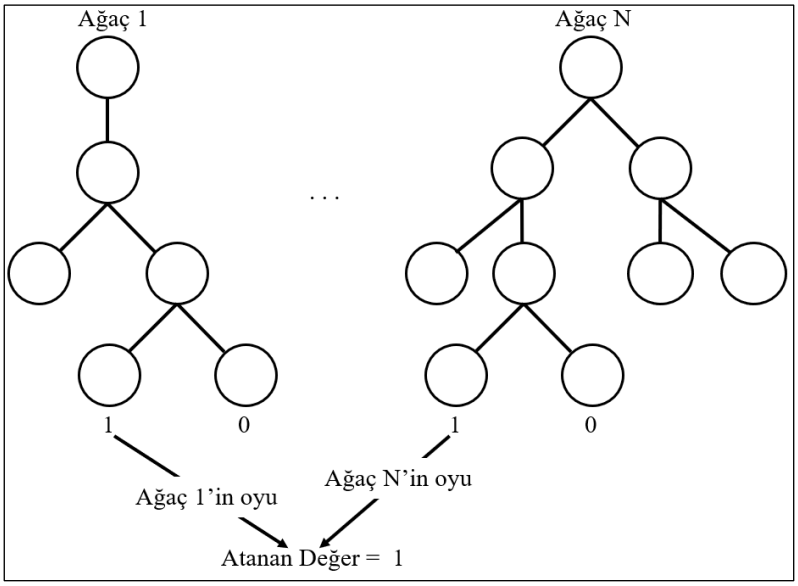
$$Entropi = -\sum_{i=1}^{m} p_i \, log_2(p_i) \tag{4.5}$$

4.1.6. Rastgele orman (RO)

Rastgele Orman (RO), birden fazla karar ağacının birleşiminden oluşan bir yapıdır. Orijinal veri içerisinden rastgele çekilmiş birbirlerinden ayrı parçalar kullanılarak, orman içerisinde bulunan ağaçların eğitilmesi esas alınmaktadır. RO içerisinde ağaçların büyümeleri esnasında rastgele özellik seçim işlemi uygulanmaktadır. Bunun da sebebi olarak, çok büyük boyutlardaki veri setleri üzerinden tek bir karar ağacının yeteri kadar

sağlıklı sonuç vermesi zor olacağından dolayı, büyük veri setinin parçalara ayrılıp ormandaki karar ağaçlarında ayrı ayrı öğrenme süreçlerinin gerçekleştirilmesi süreçlerinin, doğruluk oranlarını artırması olarak gösterilmektedir [142]. Rastgele özellik seçim süreci ile orman içerisindeki ağaçlar, değiştirme yöntemi ile eğitim özelliklerinin alt kümelerinin belirlenmesi ile oluşturulmaktadır. Bu durum, aynı özelliğin birkaç kez seçilebileceği, bazı özelliklerin ise hiç seçilemeyeceği anlamına gelmektedir [143]. Bu sayede daha hızlı ve gürültüye daha fazla dayanıklı öğrenme modelleri oluşturulabilmektedir. Buna ek olarak orman içerisindeki ağaçlar, modele esneklik kazandırdıkları için, sınıflandırma, kümeleme ve regresyon işlemlerinin performansını artırmaktadır. Özellikle büyük veri setlerinin değerlendirilmesi aşamasında, iyi bir seçim olarak değerlendirilmektedir [144].

RO, ağaç yapılı sınıflandırıcıların birleşiminden oluşan bir sınıflandırma algoritmasıdır. Bahsi geçen ağaç yapılı sınıflandırıcılar, birbirlerinden bağımsız ancak, aynı şekilde dağıtılmış rastgele vektörler içermektedir ve yapısı içerisinde bulunan her ağaç, giriş verilerindeki en popüler sınıf değeri için oy verme süreci uygulamaktadır [145]. Ağaçların kullandıkları oylar sonucunda daha fazla ağaçtan oy almış olan sınıf etiketi, RO algoritması tarafından belirlenmiş etiket değeri olarak gösterilir (Şekil 4.8) [143].



Şekil 4.8. Rastgele orman algoritması, sınıflandırma şeması [143]

4.1.7. k - En yakın komşu algoritması (k-NN)

k-NN algoritması, sınıflandırma teknikleri altında benzerlik fonksiyonlarını çalıştıran ve bu şekilde tahmin süreçlerini çalıştıran bir algoritmadır. 1950'li yıllarda keşfedilmiş ve hala günümüzde popüler olarak kullanımı devam etmektedir. Çalışma mantığına, iki boyutlu bir düzlemde bakmak gerekmektedir. Bunun için Şekil 4.9'da görüldüğü gibi veriler iki boyutlu bir düzleme yerleştirilmekte ve dikey ve yatay eksen (x ve y) değerlerine göre, benzerlik değerleri hesaplanmaktadır [125].

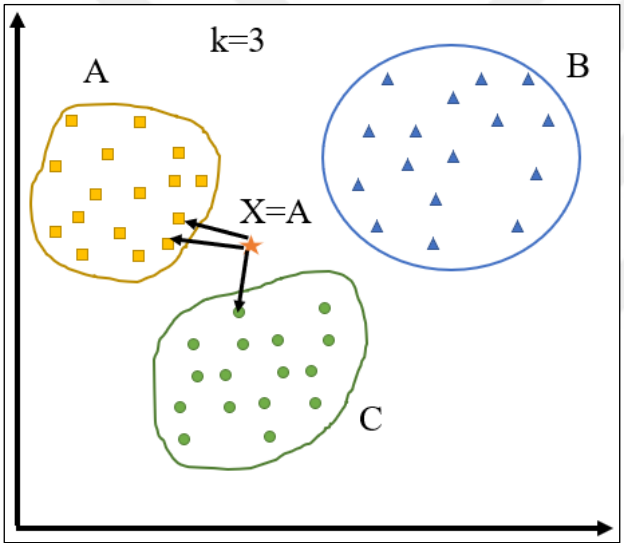
k-En Yakın Komşu Algoritması (k-NN), makine öğrenmesi algoritmalarının içerisinde en ılımlı çalışan sınıflandırıcı türüdür. k ile ifade edilen benzerlik vektörlerine dayanan çalışma sisteminde, bir nesnenin sınıflandırılması sürecinde k adet komşularının içerisindeki en çok oy alan sınıf değeri kullanılmaktadır. Sınıfı tahmin edilecek olan değerin komşularının adedini belirten k, genellikle çok küçük pozitif bir tamsayıdır [146]. k-NN algoritması, sınıf etiketi bilinmeyen yeni bir örnek veri giriş yapıldığında, bu verinin sınıflandırılma sürecinin yakın komşu konumunda bulunan çoğunluk tarafından gerçekleştirilmesi sağlayan denetimli bir algoritma olarak görev yapmaktadır. k-NN ile, test verileri ile algoritmanın sınanması süreçlerinde ve yeni girişi yapılan verinin sınıf etiketinin tespit edilmesi süreçlerinde, bu verilerin öğrenme kümesindeki veriler ile aralarındaki mesafeyi ölçerek, test edilecek veya sınıfı belirlenecek veriye en yakın olan k adet komşuyu bulmaya odaklanmaktadır. Burada bahsedilen mesafenin hesaplanması konusunda da genel anlamda Öklid eşitliğinin (Eş. 4.6) kullanılması tercih edilmektedir [147].

$$E(i,j) = \sqrt{\sum_{k=1}^{n}(i_{k-}\,_{j_k})^2} \tag{4.6}$$

k-NN, temel manada verilerin belirli bir özellik alanına ait olduğunu varsaymaktadır. Bu sebeple, yeni girişi yapılan veri noktasının, öğrenme kümesindeki tüm noktalara olan uzaklıkları tek tek dikkate alınmaktadır. Yeni girilen verinin sınıf değerinin tespit edilmesi konusunda da uzaklıkları hesaplanmış olan komşular için başta belirlenmiş olan k değeri baz alınır. Eğer k=1 olarak belirlenmişse, sadece en yakın mesafedeki komşunun sınıf değerine göre karar verilir. k değerinin optimum seviyede belirlenmesi önemlidir. Çünkü, k çok küçük olursa, algoritma öğrenme kümesinde bulunan gürültüden çok fazla

etkilenecektir. k değeri çok büyük olursa, aslında çok uzak olan komşularda yakın olarak değerlendirilecek ve bunun sonucunda verinin sınıfının tahmininde hata yapma ihtimali artacaktır. k-NN algoritmasının çalışması şu şekilde sıralanabilir [148]:

1. k değeri başlatılır,
2. Giriş verisi ve öğrenme kümesi verileri aralarındaki mesafeler ölçülür,
3. Ölçülen mesafe değerleri sıralanır,
4. k adet en yakın komşular belirlenir,
5. Basit çoğunluk sistemi ile komşular arasında en fazla bulunan sınıf değeri, giriş verisi için tahmin edilir (Şekil 4.9).



Şekil 4.9. k-NN algoritması için örnek şema [148]

4.1.8. Gradient boosting machine (GBM)

Yükseltme algoritmaları, makine öğrenmesi işlemleri için ilk etapta sınıflandırma süreçlerinde kullanılmak üzere tanıtılmıştır. Algoritmalardaki temel yaklaşım, gelişmiş seviyede doğru tahmin yapabilme kapasitesine sahip bir güçlü öğrenici oluşturabilmek için zayıf öğrenenler olarak isimlendirilen birkaç tane basit modelin yinelemeli olarak birleştirilmesidir [149].

GBM, farklı türlerde olabilen pratik uygulamalarda önemli derecede başarı gösteren güçlü makine öğrenmesi algoritmaları ailesidir. GBM uygulamalarında öğrenme süreçleri, yanıt değişkeninin doğruluk oranı daha yüksek bir tahmin değeri sağlayabilmesi için art arda

yeni modellere uyum sağlaması şeklinde gerçekleşmektedir. Bu yapının arkasında yatan ana fikir, yeni oluşturulan temel öğrenicilerin, tüm toplulukla ilişkili olan hata fonksiyonlarının negatif gradyanı ile en yüksek seviyede ilişkilendirilebilecek şekilde oluşturulmasıdır. Algoritmanın uygulanması esnasında seçilen hata fonksiyonları genel olarak, o ana kadar elde edilen birçok çeşitli hata fonksiyonları ve yapılan araştırmanın kendi yapısına özgü hata uygulama olasılığı ile araştırmacıya bağlı bulunmaktadır [150].

GBM yapısı içerisinde kullanılan gradyan ifadesi, model oluşturulduktan sonra ortaya çıkan hata veya kalıntıyı ifade etmektedir. Artırma ise sonucun iyileştirilmesi anlamına gelmektedir. Teknik olarak Gradyan Artırma Makinesi olarak isimlendirilmektedir. Gradyan artırma, oluşan hatanın kademeli olarak iyileştirilmesi, doğruluk oranının daha yüksek seviyelere ulaştırılması olarak ifade edilmektedir. Örnek olarak %80 doğruluk oranına sahip bir model üzerinden incelemek gerekirse [151]:

- İlk etapta model aşağıdaki şekilde oluşturulacaktır.
- Y = M(x) + hata
- Bu modelde Y bağımlı değişkendir, M(x) ise x bağımsız değişkenlerini kullanan karar ağacıdır.
- Sıradaki işlem, önceki karar ağacı üzerinden hatanın tahmin edilmesidir.
- hata = G(x) + hata2
- G(x), x bağımsız değişkenlerini kullanarak hatayı tahmin etmeye çalışan başka bir karar ağacıdır.
- Sonraki adımda, yine aynı işlemlere benzer olarak, yine x bağımsız değişkenlerini kullanarak hata2'yi tahmin etmeye çalışan bir model oluşturulmaktadır.
- hata2 = H(x) + hata3
- İterasyonlar tamamlandıktan sonra tüm modeller bir araya getirildiğinde ortaya aşağıdaki yapı çıkmaktadır.
- Y = M(x) + G(x) + H(x) + hata3
- Sonuç olarak tek başına kullanılan M karar ağacı, %80 doğruluk oranına sahipken, üç karar ağacının hata tahminlerinde bir araya getirilmesi ile %80'den daha yüksek bir doğruluk oranı elde etmek son derece muhtemel bir sonuç olacaktır.

GBM, tahmine dayalı çalışan ve regresyon ile sınıflandırma ağaçlarından oluşan bir yapıdır. Regresyon ve sınıflandırmanın her ikisi de kademeli olarak geliştirilmiş tahminler üreten, tahmine dayalı sonuçlar ile ileriye dönük varsayımlar oluşturmaya çalışan öğrenme yöntemleridir. Güçlendirme kavramı, ağaçların doğruluk oranını artırmaya yardımcı olmaya çalışan esnek ve doğrusal olmayan bir regresyon sürecidir. Zayıf sınıflandırma algoritmaları, bir dizi karar ağacı oluşturabilmek için artımlı olarak güncellenen verilere sırayla uygulanmaktadır. Bu şekilde bir dizi zayıf tahmin modelleri üretilebilmektedir. Ancak ağaç sayısını artırmak, doğruluk oranını yükseltirken, modelin hızını ve kullanıcı tarafından değerlendirilebilme yeteneğini de azaltmaktadır. GBM, bahsi geçen bu problemi düzeltmek için ağaçları güçlendirme tekniğini genele yaymaktadır [152]. Şekil 4.10'da GBM algoritması görülmektedir.

Gerekli: Eğitim veri seti D; GBM yineleme sayısı T

Sağlanan: Tahmin edilen işlev

1. Örnek olarak sabit değer kullanarak, $g_0(x)$ sıfır taban modelini başlatın
2. **for** t = 1, t <= T **do**,
3. Artık $q_i^{(t)}$ değerini, eğitim setinin her noktasında (i = 1,2, …., N) beklenen kayıp fonksiyonunun $L(y_i, g_t(x_i))$ türevi olarak hesaplayın.
   $$q_i^{(t)} = \left\{ - \frac{\partial L(y_i, z)}{\partial z} \Big|_{z = g_{i-1}(x_i)} \right\}$$
4. Veri kümesinde yeni bir temel model (bir regresyon ağacı) h(x) oluşturun. $\{(x_i, q_i^{(t)})\}$
5. En iyi gradyan azalmasını bulun. Adım boyutu y_t:
   $$y_t = \arg min_y \sum_{i=1}^{N} L\left(y_t, g_{t-1}(x_i) + y h_t(x_i)\right)$$
6. Tüm modeli güncelle. $g_t(x) = g_{t-1}(x) + y_t h_t(x)$
7. **end for**
8. Çıktı

$$\boldsymbol{g_M(x) = \sum_{t=1}^{T} y_t\, h_t(x) = g_{t-1}(x) + y_t\, h_t(x)}$$

Şekil 4.10. GBM algoritması [153].

Makine öğrenmesi süreçlerinde, eğitim verileri üzerinde yaşanan aşırı öğrenme durumu, GBM için önemli bir endişe kaynağı oluşturmaktadır. Aşırı öğrenme, öğrenme modelinin tahmin doğruluk oranının azalması pahasına eğitim verilerine çok iyi uyum sağlaması eğilimidir. Bu durum, tahmin edilen modelin yeni karşılaşılacak veriler için genellenemeyeceği anlamına gelmektedir. Bu durumun sebebi olarak, aşırı öğrenme durumuna düşen bir modelin, girdi çıktı ilişkisinin genel bir modelini oluşturma yerine, eğitim verilerinde var olabilecek olan gürültülü verilerin özelliklerine uyum sağlamış olması gösterilmektedir. GBM tarafından oluşturulmuş olan bir modelde, eğer bir

uygulayıcı model kapsamında çok fazla yineleme seçerse veya karar ağaçlarının çok büyük bir derinliğini seçerse aşırı öğrenme sorunu ortaya çıkabilmektedir [149].

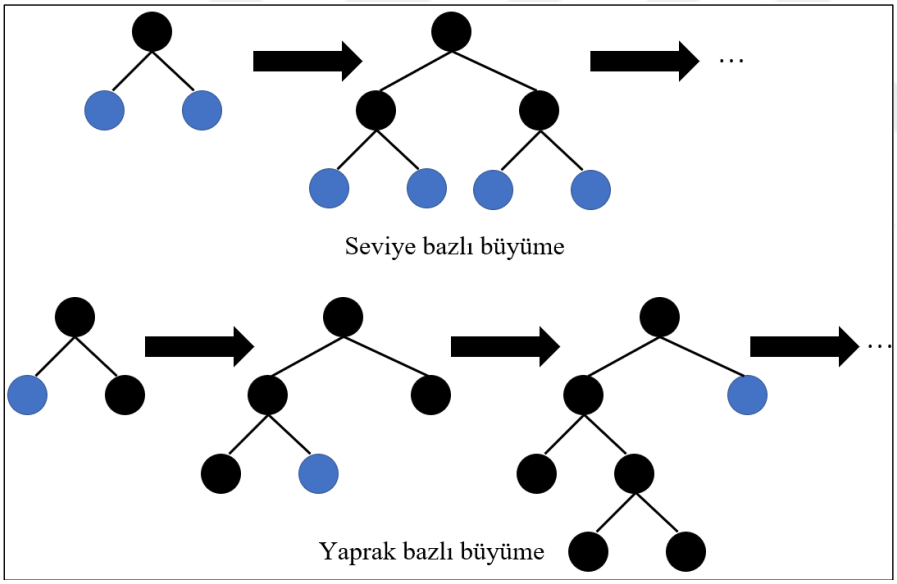
4.1.9. Light gradient boosting machine (LGBM)

LGBM, karar ağaçları ve zayıf öğrenenler teorisine dayalı gradyan öğrenme yapısının güçlendirilmiş bir versiyonudur. 2017 yılında geliştirilmiş olduğu zamandan bu yana, tahmin doğruluğunun yüksekliği, çok daha hızlı hesaplama kabiliyeti ve aşırı öğrenme problemlerini en az seviyeye indirebilme konusundaki mükemmel yeteneği sayesinde günümüzde birçok alanda başarı ile uygulanmaktadır [154]. LGBM ile büyük boyutlu veri setleri üzerinde verimlilik ve ölçeklenebilirlik problemleri çözülebilmektedir. Ayrıca LGBM, tahminlerdeki başarım oranını daha üst seviyelere çıkarabilmek için yaprak bazlı bir strateji benimsemektedir [155].

LGBM, karar ağaçlarını dikey yönlü büyümesinin gerçekleştirilebilmesi amacıyla yaprak şeklinde bir algoritma kullanmaktadır. Burada ağacın kesilebilmesi ve büyütülebilmesi için kayıp değeri en düşük seviyede bulunan yaprak seçilmektedir. Eğitim modelinin daha iyi geliştirilebilmesi için LGBM tarafından bulunan veri örneklerinin önem derecelerinin belirtilmesinde gradyan tabanlı tek taraflı örnekleme algoritması kullanılmaktadır. Bu algoritmanın ana fonksiyonu, yüksek gradyan seviyesine sahip veri örneklerine odaklanarak, düşük seviyeli gradyana sahip veri örneklerini görmezden gelmesi olarak ifade edilmektedir. Çünkü düşük seviyeli gradyana sahip veriler, aynı seviyede düşük hata oranına sahip oldukları ve bu sebeple iyi eğitilmiş oldukları varsayılmaktadır. Bundan dolayı kullanılan örnekleme algoritması, bahsi geçen bu az bilgilendirici veri noktalarını yok saymayı ve en iyi bölünme düğümlerinin hesaplanmasında geri kalan verileri kullanmayı tercih etmektedir. Tabi bu durumun sonucu olarak yüksek seviyeli gradyana sahip veriler için eğitim modelinde ön yargılar oluşacağından dolayı gradyan tabanlı tek taraflı öğrenme algoritması, yüksek seviyeli gradyana sahip veri noktalarını elinde tutarken, düşük seviyeli gradyana sahip veriler için de rastgele bir örnekleme gerçekleştirerek, ağırlık oranlarını artırmaya gitmek suretiyle sorunu çözmeye çalışacaktır [156].

LGBM, ağaç tabanlı makine öğrenmesi algoritmaları içerisinde çalışan bir gradyan artırma çerçevesidir. LGBM'nin avantajları içerisinde, hızlı eğitim süreci, yüksek verimli öğrenme

modeli, düşük seviyeli bellek kullanımı ve yüksek tahmin doğruluğu, paralel öğrenme desteği ve büyük boyutlu veri setleri işleyebilme kabiliyeti gösterilmektedir. LGBM, karar ağacına dayalı topluluk yöntemleri içerisinde farklı bir yapıya sahiptir. Karar ağacına dayalı öğrenme algoritmalarının çoğunluğu Şekil 4.11'in üst bölümünde görüleceği üzere ağacı seviye bazında büyütme yoluna gitmektedirler. LGBM ise Şekil 4.11'in alt bölümünde gösterildiği şekilde diğer algoritmaların aksine yaprak bazlı ağaç büyütme tekniğini kullanmaktadır. Bu sayede çok daha karmaşık olan problemlerin üstesinden gelebilmektedir. LGBM ağacın büyütme aşamasında en yüksek delta kaybına sahip niteliği seçtiği için, seviye bazlı algoritmalardan daha düşük kayıp elde edebilmektedir. LGBM ayrıca, gelecekteki öğrenme süreçleri için düşük gradyanlı verilerin kullanımının azaltan gradyan tabanlı tek taraflı örnekleme ve özellikleri bir araya getiren özel özellik gruplandırma araçları ile karakterize edebilmektedir [157] [158].

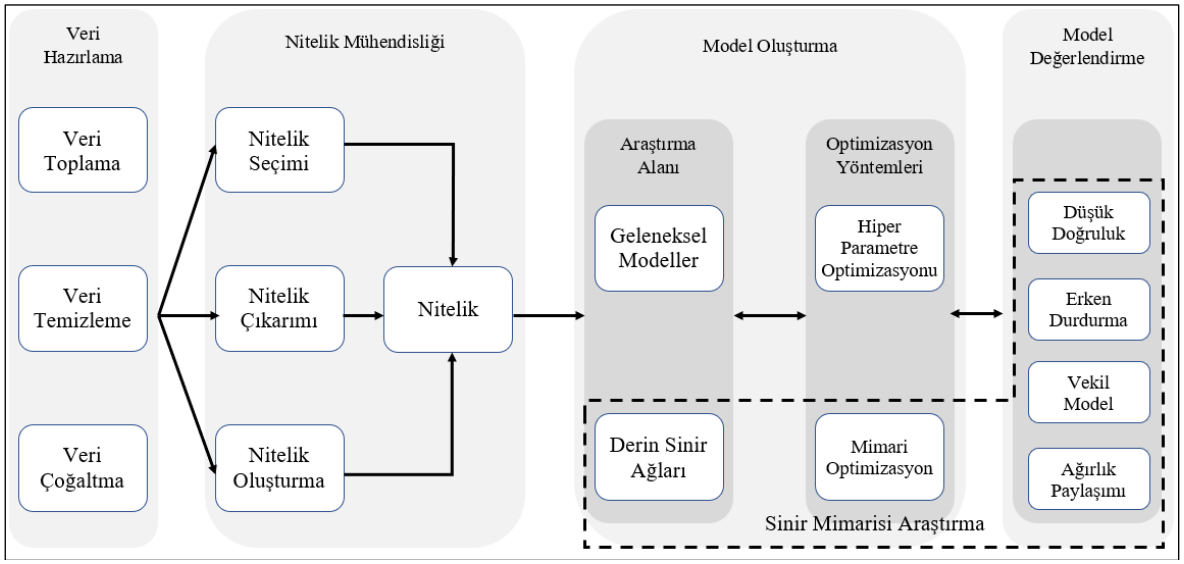


Şekil 4.11. Karar ağacı öğrenme algoritmalarında kullanılan karar ağaçlarını büyütmenin farklı yolları [158]

4.2. Otomatik Makine Öğrenmesi (AutoML)

Makine öğrenmesi algoritmalarının kullanıma hazırlanması, gerekli ayarlamalarının yapılması, uzmanlık isteyen ve zaman alan bir süreçtir. AutoML sistemlerinin geliştirilmesinin sebebi de bu süreçlerin basitleştirilmesi ve öğrenme modellerinin daha hızlı ve kolay bir şekilde oluşturulmasının sağlanmasıdır. Veri analiz işlemleri konusunda yetkin olmayan kişilerin, verimli bir şekilde makine öğrenmesi yapılarını kullanmak amacıyla, uzmanların ise özellikle hiper parametrelerin seçimlerinin otomatikleşmesi ile

daha hızlı veri analizi yapmaları amacıyla AutoML kullanımı tercih edilmektedir [159]. AutoML yapılarının oluşturulmasının sebeplerinden biri de uzmanların istatistiksel ve makine öğrenmesi bilgilerine ve tecrübelerine çok fazla ihtiyaç duymadan öğrenme modellerini otomatik bir şekilde oluşturmalarını sağlamaktır [160]. Şekil 4.12'den görüleceği üzere AutoML, veri hazırlama, nitelik mühendisliği, model oluşturma ve model değerlendirme olmak üzere birkaç sürecin ardışık bir şekilde gerçekleşmesinden oluşmaktadır [161].



Şekil 4.12. AutoML ardışık düzeni [161]

AutoML düzeninde birinci basamak veri hazırlama aşamasıdır. Makine öğreniminde toplanan verilerin ilk halleri üzerine önişleme çalışmalarının gerçekleştirilmesi ile kaliteli verilerin oluşturulması gerekmektedir. Pratikte, veri önişleme ve veri hazırlama aşamaları makine öğrenim sürecinin %80'ini oluşturmaktadır. Bundan dolayı veri hazırlama makine öğrenmesi için önem arz eden bir konudur [162]. AutoML ardışık düzeninde veri hazırlama, veri toplama, veri temizleme ve veri çoğaltma aşamalarından oluşmaktadır. Veri toplama, yeni bir veri seti oluşturmak veya var olan veri setini genişletmek için kullanılması gereken bir süreçtir. Veri temizleme sürecinde, veri içerisinde olabilecek gürültüler, eksiklikler, hatalar ve sapan veri gibi olumsuzlukların filtrelenmesi sağlanmaktadır. Bu sayede öğrenme modeli performansının düşmesi engellenmeye çalışılmaktadır. Veri çoğaltma süreci ise, oluşturulan öğrenme modelinin sağlamlığını ve performansını artırmak için kullanılmaktadır [161].

AutoML sürecinin ikinci aşaması olan nitelik mühendisliğinde, ham veriler üzerinde gerekli kümeleme ve analiz süreçlerin gerçekleştirilmesi ile niteliklerin oluşturulması ve bu niteliklerin makine öğrenmesi modelleme süreçlerine uygun hale getirilmesi işlenmektedir. Bu adım, makine öğrenmesi süreçleri için büyük önem arz etmektedir. Çünkü doğru niteliklerin belirlenmesi ve doğru bir şekilde düzenlenmesi, öğrenme modelinin oluşturulmasını kolaylaştırabilir ve modelin başarımını üst seviyelere çıkartabilir [163]. AutoML yapısında nitelik mühendisliği, nitelik seçimi, nitelik çıkarımı ve nitelik oluşturma olmak üzere 3 alt bileşenden oluşmaktadır. Nitelik seçimi, makine öğrenmesi süreçleri içerisinde gerçekleştirilen önemli görevlerden biri olarak tanımlanmaktadır. Nitelik seçimi, gereksiz ve ilişkisiz verilerin çıkartılması ile veri setinde boyut küçültme işleminin uygulanabilmesi amacıyla sıklıkla kullanılmaktadır. Ayrıca birbirleri ile en ilişkili olan niteliklerin belirlenmesi ile daha başarılı modellerin oluşturulmasına imkan sağlamaktadır [164]. Nitelik çıkarımı, veri seti üzerindeki başarımı artırmak amacıyla, veri seti boyutunun küçültülmesi için kullanılan tekniklerden biridir. Veri içerisinde bulunan daha fazla bilgiye ulaşabilmek için fonksiyonel haritalama yolu ile orijinal niteliklerden alt küme şeklinde yeni niteliklerin çıkarılmasını sağlamaktadır [165]. Nitelik oluşturma süreci ise, nitelikler arasında var olan gizli ilişkileri ortaya çıkartmak ve yeni bileşik niteliklerin oluşturulmasını amaçlayan bir süreçtir [166]. Nitelik oluşturma süreci, orijinal niteliklerin temsil yeteneklerini daha da iyi seviyelere ulaştırmak için yapılmaktadır. Genel anlamda uzman kişiler tarafından uygulanmasına ihtiyaç duyulmaktadır. Nitelik oluşturma sürecinde, standardizasyon, normalizasyon ve nitelik ayrıklaştırma, genellikle kullanılan yöntemler arasında sayılmaktadırlar [161].

Nitelik mühendisliği sürecinde işlemler tamamlandıktan sonra artık öğrenme modelinin oluşturulması gerekmektedir. Model oluşturma süreci temelde araştırma alanı ve optimizasyon yöntemleri olmak üzere ikiye ayrılmaktadır [161]. AutoML yapısında hiper parametrelerin seçimi ve mimariler arasında karar verilmesi bir araştırma problemi olarak tanımlanır ve bu problemin çözümü için tüm seçeneklerin temsil edildiği ve değerlendirildiği bir araştırma alanı oluşturulur. Araştırma alanı, doğru hiper parametrelerin tespit edilmesi amacıyla test yapılacak olan parametre aralıklarını belirler. Mimari yapı seçimleri için ise olası yapıların hangileri oldukları yine araştırma alanı tarafından belirlenmektedir. Araştırma alanı kendi içerisinde DVM, LR, k-NN gibi geleneksel modellerden ve derin sinir ağları modellerinden oluşmaktadır [167]. Geleneksel modeller ile derin sinir ağları modellerini karşılaştırmak amacıyla yapılmış olan bazı

çalışmaların sonuçlarına göre, derin sinir ağlarının başarım oranı, geleneksel modellere daha iyi seviyelere ulaşmaktadır [168] [169].

Model oluşturma sürecinin ikinci aşaması olan optimizasyon yöntemleri de hiper parametre optimizasyonu ve mimari optimizasyon olmak üzere iki seçenekten oluşmaktadır. Optimal başarı elde eden bir makine öğrenmesi öğrenme modeli oluşturabilmek için en uygun hiper parametre konfigürasyonu yapabilmek önem arz etmektedir. Hiper parametre konfigürasyonu kategorik, ayrık ve sürekli hiper parametreler olmak üzere makine öğrenmesi algoritma türlerine göre farklılıklar göstermektedir. Hiper parametrelerin manuel konfigürasyonu, geleneksel olarak yoğun bir şekilde kullanılmaktadır. Ancak konfigürasyon sürecinde kullanılabilecek çok sayıda hiper parametre olması, karmaşık makine öğrenmesi modellerinin kullanılması ve test süreçlerinin zaman alması sebeplerinden dolayı AutoML yapısı içerisinde bulunan hiper parametre optimizasyonunun kullanılması öne çıkmaktadır. Hiper parametre optimizasyonunun temel amacı, hiper parametre konfigüre etme sürecini otomatikleştirmek ve kullanıcıların makine öğrenimi pratik problemlere etkili bir şekilde uygulamalarını mümkün kılmaktadır [170]. Diğer optimizasyon seçeneği olan mimari optimizasyon, araştırma alanının tanımlanmasından sonra çalışmaya ve yüksek başarım sağlayacak eğitim modelini oluşturacak olan en iyi mimariyi aramaya başlar. Genel anlamda bir sinir ağı mimarisi test veri setinde ölçülen performans değerine göre konfigüre edilmiş olan statik hiper parametreler olarak düşünülmektedir. Ayrıca bu sürecin tamamlanması, önemli derecede uzmanlık gerektirmekte ve büyük oranda insan faktörüne bağlılık göstermektedir. Bundan dolayı deneme yanılma yöntemi ile en iyinin tespit edilmesi gerektiğinden dolayı çok fazla iş gücü ve zaman ihtiyacı bulundurmaktadır. Bu sebeplerden dolayı süreci daha hızlı ve kolay bir hale getirmek ve otomatik olarak en iyi mimarinin aranmasını sağlamak amacıyla mimari optimizasyon süreçlerinin kullanımı önerilmektedir [161].

AutoML ardışık düzenindeki son aşama, modelin değerlendirilmesi sürecidir. Bu süreç, kendi içerisinde düşük etkileşim, erken durdurma, vekil model ve ağırlık paylaşımı bölümlerinden oluşmaktadır [161]:

- Düşük Etkileşim**:** Makine öğrenmesi içerisinde oluşturulan öğrenme modelinin eğitim süresi, veri setinin boyutu ve oluşturulmaya çalışılan modelin yapısı ile yakından ilişkilidir. Bu sebeple oluşturulan öğrenme modelinin test edilmesi

sürecinin, veri seti boyutunun düşürülmesi gibi çalışmalarla kısaltılması ile ilgili işlemler bu aşamada tanımlanmaktadır.

- Erken Durdurma**:** Öğrenme modelinin test edilmesi esnasında oluşacak aşırı uyum sebebiyle test sürecinin uzamasını engellemek için bazı değerlendirmelerin iptal edilmesi işlemleri bu aşamada gerçekleştirilmektedir.
- Vekil Model**:** Öğrenme modellerinin performanslarını tespit etmek amacıyla kullanılan güçlü bir araçtır. Vekil model, çok az sayıda etiketli veriler kullanılarak eğitimden geçirilir. Sonrasında ise modelin değerlendirilmesi yapılmadan doğrudan tahminlerde bulunmak amacıyla kullanılır.
- Ağırlık Paylaşımı**:** Sinir ağlarında, ağ tasarım süreçlerinin ve ağ arama işlemlerinin zamanını kısaltmak amacıyla kullanılan bir tekniktir. Model için üretilen ağırlık değerleri, oluşturulan alt ağlar arasında paylaştırılarak iş bölümünün yapılması ve bu sayede daha hızlı işlem gerçekleştirilmesi sağlanmaktadır.

Çalışma kapsamında değerlendirmek üzere, Auto-Keras, Auto-Sklearn ve PyCaret kütüphaneleri tercih edilmiştir. Auto-Keras ve Auto-Sklearn kütüphaneleri, genel anlamda diğer kütüphanelere göre daha iyi performans sergilediği çeşitli çalışmalarda belirtildiği için tercih edilmiştir [171] [172] [173]. Düşük kodlu olarak ifade edilen AutoML kütüphaneleri tarafından oluşturulabilecek sonuca yönelik iş örneklerinin artması, makine öğrenmesi alanlarında farkındalığı artırmak için önemlidir. Bundan dolayı, bu çalışmada üçüncü AutoML kütüphanesi olarak düşük kodlu bir yapıya sahip olan PyCaret tercih edilmiştir [174].

4.2.1. Auto-Keras kütüphanesi

Auto-Keras, diğer AutoML kütüphanelerinde olduğu gibi, makine öğrenimi teknolojilerine yeterince hakim olmayan kişilerin de kolaylıkla veri analizleri yapabilmelerine imkan sağlamak amacıyla geliştirilmiş, açık kaynak kodlu bir kütüphanedir. Ancak Auto-Keras, diğer kütüphanelere nazaran, çoğunlukla derin öğrenme modellerine odaklanmaktadır [172].

Auto-Keras kütüphanesinin kullanım amaçları, öğrenme modelini elde edecek en uygun mimarinin tespit edilebilmesi için otomatik arama fonksiyonlarının çalıştırılması ve analiz sürecinde kullanılacak olan derin öğrenme algoritmaları için gerekli hiper parametre

seçimlerinin otomatik olarak yapılması olarak belirtilmektedir. Makine öğrenmesi süreçlerini basitleştirmek için geliştirilmiş olan Auto-Keras, bunun için sinir mimarisi arama (Network Architecture Search - NAS) algoritmalarından yararlanmaktadır [175].

Auto-Keras, Keras modülüne dayalı ve sinir mimari araması (NAS) mantığına göre çalışan bir kütüphanedir. Auto-Keras içerisinde kullanılan NAS yapısı, yoğun katmanların, birimlerin, uygulama yapılan aktivasyon fonksiyonları türünün, bırakmak değerlerinin ve diğer derin öğrenme hiper parametrelerinin değerlerini otomatik ayarlamak için kullanılmaktadır. Bu düzenlemeleri gerçekleştirmek için de Bayes Optimizasyonundan faydalanmaktadır [176].

NAS sisteminin arkasında çalışan fikir, herhangi bir analiz uzmanı tarafından müdahaleye gerek kalmadan mümkün olan en iyi şekilde sinir ağı mimarisinin otomatik olarak tasarlanabilmesidir. NAS ile sinir ağı mimarisinin oluşturulmasında iki yöntemin çoğunlukla kullanıldığına literatürde rastlanmaktadır. İlk yöntem, gradyan metodunu kullanarak sıfırdan evrişimli sinir ağı mimarisinin oluşturabilen pekiştirmeli öğrenme yapısıdır. Bu yöntemde doğruluk değeri verilir ve sonrasında kontrol mekanizmasını güncellemek için gradyan hesaplamaları kullanılır. Bu şekilde aynı anda yeni hiper parametrelerin değerleri de ayarlanabilmektedir. İkinci yöntem, en iyi sonucu elde edebilmek için önceden eğitilmiş olan ağ mimarisinin kullanılmasıdır. Bu yöntemde alternatif olarak genetik algoritmalar kullanılabilmektedir [177].

4.2.2. Auto-Sklearn kütüphanesi

Auto-Sklearn, bir eğitim veri seti için en uygun öğrenme modelinin oluşturulması amacıyla, meta öğrenme ve sıralı model tabanlı algoritma düzenleyici (Sequential Model-Based Algorithm Configurator - SMAC) yöntemlerini kullanmaktadır. Kullanıcı tarafından yüklenen veri seti üzerinde makul bir öğrenme modeli önerdikten sonra, modelin değerlendirilebilmesi amacıyla meta öğrenme yapısı ile SMAC araması başlatmaktadır. Bu şekilde yinelemeli SMAC aramaları ile önerilen modellerin testleri yapılarak en iyi performansı gösteren model tespit edilir ve Bayes Yaklaşımı kullanılarak en iyi model kullanıcıya döndürülür [178].

Auto-Sklearn, Scikit Learn üzerine oluşturulmuş bir kütüphanedir. Diğer tüm AutoML kütüphaneleri gibi en iyi öğrenme modelini oluşturan algoritmanın ve gerekli hiper parametrelerin otomatik seçilmesinin sağlanması ile makine öğrenmesi süreçlerinin daha hızlı ve kolay bir şekilde gerçekleştirilmesini amaçlamaktadır. Auto-Sklearn, ardışık düzen kurgusuna sahiptir ve bu kurguyu optimize etmek için Bayes yaklaşımını kullanmaktadır. Bu sayede hiper parametre organizasyonlarının yapılması süreçlerinde Bayes yaklaşımı üzerinden gerekli parametre dengelemelerini yapabilmektedir [175].

Auto-Sklearn, özellik seçiminin tamamen otomatikleştirilmesini sağlamaktadır. Girdi olarak verilen veri setine en uygun makine öğrenmesi algoritmasının seçilmesi ve seçilen algoritma için gerekli hiper parametre ayarlarının ve değerlerinin otomatik bir şekilde belirlenmesini sağlar. Çok sayıda çeşitli veri setleri üzerinde yapılan deneylerde başarılı olduğu kanıtlanmıştır. Bu sayede hem uzmanlar hem de acemiler tarafından etkili ve verimli öğrenme modelleri oluşturmak amacıyla kullanılmaktadır [179].

Auto-Sklearn, öğrenme modellerinin oluşturulması amacıyla, 15 adet makine öğrenmesi algoritması, 14 adet önişleme metodu ve 4 adet veri önişleme tekniği kullanarak karşılaştırmalar yapmaktadır [173].

4.2.3. PyCaret kütüphanesi

PyCaret, diğer tüm AutoML kütüphanelerinde olduğu gibi, makine öğrenmesi süreçlerinin uzmanların haricinde de kısa sürede başarılı bir şekilde gerçekleştirilebilmesi amacıyla geliştirilmiş olan ve diğerlerinden daha az kod kullanan, açık kaynak kodlu bir AutoML kütüphanesidir [180]. Diğer açık kaynaklı makine öğrenimi kütüphanelerine göre PyCaret, çok fazla kod satırını birkaç farklı kelime ile değiştirmek amacıyla kullanılabilecek bir kütüphane sunmaktadır. Bu sayede öğrenme modellerinin oluşturulması konusundaki deneyler daha hızlı ve daha verimli bir şekilde gerçekleştirilmektedir. PyCaret, ilk defa Gartner tarafından dile getirilmiş olan Vatandaş Veri Bilimcisi kavramı temel alınarak tasarlanmıştır. Vatandaş Veri Bilimcisi kavramı da uzman veri analistlerine ulaşmanın zorluğu karşısında, yaratılmış özel sistemler ile uzman olmayan kişilerin de veri analizi yapabileceklerini işaret etmektedir. PyCaret bu çerçeveye göre uyarlandığından dolayı, sade ve kolay bir yapıya kavuşmuştur [181]. PyCaret, öğrenme modellerini oluşturmak amacıyla 18 farklı makine öğrenmesi sınıflandırma algoritması üzerinden model üretip

bunların karşılaştırmalı çizelgelerini kullanıcıya sunmaktadır. Genel anlamda, kullanıcının belirlediği çeşitli ölçütlere uygun olarak tercih edilen modellere vurgu yapabilmektedir. Ayrıca ürettiği çizelgelerde elde ettiği en önemli özellikleri de kullanıcıya göstermektedir [174].

PyCaret, sınıflandırma işlemlerinde kullanılan Python tabanlı makine öğrenmesi modülüdür. Kullanımının son derece kolay olması ve herkes tarafından rahatlıkla uygulanabilmesi sayesinde son zamanlarda popülerliği artmaktadır [182].

PyCaret, yapısı gereği düşük kodlu bir AutoML kütüphanesi olarak tasarlanmıştır. Düşük kod demek, bir kullanıcının herhangi bir makine öğrenimi modelini oluşturabilmek için önemli derecede daha az satır kod yazması gerekeceği anlamına gelir. Bu sayede kullanıcıların yüzlerce satır kod yazmalarına gerek kalmamaktadır. PyCaret kütüphanesinin ana kullanım amacı, veri analizi konusunda uzman olmayan araştırmacıların da makine öğrenmesi tekniklerinden tam anlamı ile faydalanabilmesidir. Aynı zamanda uzmanlara da çok daha az çaba ile en iyi öğrenme modelini ve en iyi parametreleri belirlemeleri konusunda olanak sağlanmaktadır [181] [183].

4.2.4. Diğer AutoML Kütüphaneleri

Automl, farklı kişi veya kuruluşlar tarafından yazılmış birçok kütüphane yapısına sahiptir. Bu tez kapsamında değerlendirme işlemleri için tercih edilmemiş olan AutoML kütüphanelerine bu başlık altında değinilmiştir.

MLBox: Verilerin kullanıma hazır hale getirilebilmeleri amacıyla, veri temizleme ve önişleme adımlarında analizciye destek sağlayan bir kütüphanedir. Ayrıca veriler üzerinde özellik birleştirme ve ölçekleme gibi ilkel mühendislik tekniklerini de kullanmaktadır [184].

TPOT: Optimizasyon işlemlerinin gerçekleştirilmesinde genetik algoritma tekniklerini kullanan bir kütüphanedir. Sadece Scikit-Learn tabanlı öğrenme modelleri için optimizasyon desteği sağlamaktadır [185].

H2O: Denetimli öğrenme modellerinin oluşturulması sağlayan tam otomatik bir kütüphanedir. Denetimli regresyon, ikili sınıflandırma ve çok sınıflı sınıflandırma modellerinin oluşturulmasını destekler. Kütüphane tamamıyla otomatik çalışmasına rağmen, belirlediği özellikleri kullanıcıya parametre olarak sunar ve özelleştirilmesine imkan sağlar [186].

Cloud AutoML: Google tarafından geliştirilmiştir. Görüntü sınıflandırma ve nesne seçimi konularında ayarların otomatik yapılmasını sağlayan ve bulut bilişim üzerinden hizmet veren bir kütüphanedir [187].

DataRobot: Model oluşturma aşamasında H2O parametrelerini entegre ederek çalışan ticari bir kütüphanedir. Kullanıcıların otomatik model oluşturma sürecinin performanslarını ve verimliliklerini kontrol edebilmeleri için lider tablosu yapısını kullanmaktadır [188].

Auto-PyTorch: Sinir ağları parametrelerini ve eğitim süreci için gerekli hiper parametre değerlerini birlikte düzenleyebilmek amacıyla çoklu aslına uygunluk optimizasyonu kullanan bir otomatik derin öğrenme kütüphanesidir. Veri önişleme, sinir mimarisi, eğitim teknikleri ve düzenleme yöntemlerini içeren bir derin öğrenme sürecini otomatik olarak gerçekleştirir [189].

AutoGluon: Tablo halinde bulunan veri setleri için üretilmiş ve sinir mimarisi arama yönetmini kullanan bir kütüphanedir. Temel özellik olarak uygulama kolaylığı sağlar. Birleştirme ve düzenleme işlemlerini bir arada kullanarak öğrenme modellerini oluşturur. Sınıflandırma ve regresyon işlemlerinde kullanılabilir [184].

Auto-Weka: Birleşik algoritma seçimi ve hiper parametre optimizasyonu problemlerini çözebilmek amacıyla geliştirilmiş, açık kaynak kodlu ve Weka arayüzünü kullanan bir kütüphanedir. Auto-Sklearn'de olduğu gibi SMAC yapısını kullanır. Bu sebeple Auto-Sklearn ile benzer çalışma mantığına sahiptir [190].

5. İLGİLİ ÇALIŞMALAR

Zeki fabrikalar, araştırmacılar için büyük öneme sahip bir alan sunmaktadır. Birbirine bağlı makinelerin, ağ yardımı ile birbirleri ile konuşmaları şeklinde otonom bir yapıda üretim süreçlerini gerçekleştirmeleri son derece çekici bir konudur. Ayrıca, bu sistem içerisinde ortaya çıkabilecek olan hataların ve güvenlik endişelerinin de yine otonom olarak takip edilmesi ve herhangi bir anomaliye hızlı bir müdahalenin gerçekleştirilmesi de yapının kendisi kadar ayrı bir öneme sahiptir. Bu sebeple birçok araştırmacı, zeki üretim sistemleri içerisinde anomali tespiti hakkında çalışmalar yürütmektedir.

Hranisavljevic vd. 2016 yılında gerçekleştirdikleri çalışmada, hibrit üretim sistemlerinde kullanılabilecek olan otomatik öğrenme özelliğine sahip anomali tespit algoritması önermişlerdir. Sistem içerisinde uygulanan gözlemlerden tespit modelini oluşturmak için derin öğrenme teknikleri ve zamanlı otomata sistemlerinin birleşiminden yararlanmışlardır. İki adet gerçek sistem dahil olmak üzere çeşitli veri setleri üzerinde test ettikleri algoritmanın umut verici sonuçlar verdiğini açıklamışlardır [191]. 2017 yılında Birgelen ve Niggeman tarafından yayınlanmış olan devam çalışmasında da hibrit üretim sistemlerinde anomali tespiti için, denetimsiz ve parametrik olmayan bir yaklaşım oluşturulmuştur. Bu yaklaşım ile normal şartlarda anomali tespit uygulaması mümkün olmayan üretim sistemlerinde, hibrit zamanlı otomata kullanımına izin vererek anomali tespiti yapılmasını sağlayan, kendi kendini düzenleyen haritalar ve havza dönüşümleri kullanılmıştır [192].

H. Wu ve arkadaşları tarafından yapılan çalışmada, anomali tespitinde kullanılan akustik verinin ayrıştırılabilmesi için mikro hizmetlere uygun bir ağ içi çözüm önerisi geliştirilmiştir. Çalışma sonucunda veri ayrımı için harcanan sürenin %43,75 oranında azaltılabildiği tespit edilmiş ve uygulamanın kritik görevler için nasıl daha iyi kullanılabileceğini tanımladığı ifade edilmiştir [104]. Seo et al. tarafından yapılan çalışmada ise bir üretim sisteminde çalışan makineler tarafından ortaya çıkan seslerin akustik sinyaller halinde alınarak spektogram görüntülerine dönüştürülmesi ve elde edilen bu görüntüler üzerinden evrişimsel sinir ağları algoritması ile anomali tespit sistemi geliştirilmiştir. Gerçek bir fabrika ortamında denenen modelin %99,44 oranında doğru tespit yaptığı ve bu sayede makine arızalarının tespitini hızlandırarak yaralanmaların azalmasına yardımcı olduğu belirtilmiştir [193].

Lam ve Abbas tarafından 5G ağlarında anomali tespiti üzerine yapılmış olan çalışmada, ağda gerçekleşebilecek siber saldırıların tespitine yönelik, anomali tespit çalışması yapılmıştır. Çalışmada araştırmacılar, anomali tespiti için sinir arama mimarilerine dayanan Google AutoML Vision ve Google AutoML Vision Edge platformlarını kullanmışlardır. KDD CUP99 veri seti üzerine yaptıkları çalışma sonucunda, oluşturdukları öğrenme modeli, normal verilerin tespitini %100, anomali verilerinin tespitini ise %96,4 doğruluk oranı ile tamamlamıştır. Ancak çalışma kapsamında oluşturulan öğrenme modeli, veri seti içerisinde anomali verileri farklı siber saldırı türleri olarak bulunmasına rağmen, alınan sonuçlarda ilgili verinin sadece anomali olup olmadığına bakılmıştır [194].

Truong ve arkadaşları, AutoML kütüphanelerinin belirli bir veri seti üzerindeki performanslarını değerlendirmek amacıyla bir çalışma yaptılar. Çalışma kapsamında H2O-AutoML, Auto-Keras, Auto-Sklearn, Ludwig, Darwin, TPOT ve Auto-ML kütüphaneleri arasında karşılaştırma işlemlerini uyguladılar. Sonuç olarak, belirli bir veri seti üzerinde herhangi bir AutoML kütüphanesinin diğerlerine göre bireysel anlamda belirgin bir üstünlük göstermediğini belirtmişlerdir. Ancak, H2O-AutoML, Auto-Keras ve Auto-Sklearn kütüphanelerinin diğerlerine göre genel anlamda biraz daha iyi performans sergilediğini tespit edebilmişlerdir [171]. Çalışma kapsam olarak belirli kütüphanelerin karşılaştırılması üzerine kurulmuş ve AutoML kütüphanelerinin başarımını ortaya koymuştur. Ancak akan veri üzerinde gerçek zamanlı anomali tespiti için test yapılmamıştır.

Başka bir çalışmada ise zeki ulaşım, zeki enerji ve zeki fabrika sistemleri için ayrı ayrı anomali tespiti çalışması üzerinde odaklanılmıştır. Çalışmada anomali tespit süreçleri grafik sinir ağları algoritması tercih edilmiş ve elde edilen sonuçların, ilgili konuda gelecek çalışmalar için rehberlik edebileceği belirtilmiştir [195].

Pahl ve Aubet tarafından 2018 yılında yapılmış olan bir çalışmada, IoT sisteminde hizmet içi servislerin iletişim ve çalışma yapısını öğrenen ve kendini sürekli güncel tutan, kaynak verimli bir yaklaşım önerilmiştir. Önerilen bu yaklaşımın, düğümler arasındaki süreç iletişiminde akan verileri analiz ederek, öğrenmiş olduğu model doğrultusunda anomali tespiti yapabildiğini belirtmişlerdir. Çalışma sayesinde IoT sistemlerinin güvenlik seviyelerinin daha üst noktalara ulaşabildiği sonucuna varılmıştır [196].

Hasan vd. tarafından 2019 yılında yayınlanmış olan bir çalışmada, nesnelerin interneti üzerine gerçekleşebilecek siber saldırılar dolayısı ile oluşabilecek anomalilerin tespit edilmesi noktasında çeşitli makine öğrenmesi algoritmalarının performans karşılaştırmaları yapılmıştır. Yapılan ölçümler sonucunda, Karar ağacı, rastgele orman ve yapay sinir ağları algoritmalarının %94 gibi yüksek başarıya ulaştığı, performans bakımından ise rastgele orman algoritmasının öne çıktığı tespit edilmiştir [105].

Bir başka çalışmada, Hsieh vd. tarafından, üretim sistemi üzerinden bulunan cihazlardan toplanan gerçek veriler kullanılarak zeki üretim sistemlerin anomali tespiti için kullanılabilecek bir algoritma önermişlerdir. Üretim hattından elde ettikleri çok değişkenli sensor veri setlerinde bulunan sınırlı ve düzensiz anomali verilerinin ortaya çıkartılabilmesi için, otomatik kodlayıcıya dayalı denetimsiz gerçek zamanlı anomali algılama algoritması kullanmışlardır. Sonuç olarak önerdikleri bu algoritmanın anomali tespitinde %90 başarım oranı elde ettiğini belirtmişlerdir [106].

Wang vd. tarafından anomali tespiti için derin öğrenme yapıları üzerine yapılan çalışmada araştırmacılar, derin öğrenme tabanlı oluşturulan anomali tespit sistemlerinin daha iyi anlaşılmasını amaçlamışlardır. Çalışmada ilk etapta derin öğrenme yapılarından önce uygulanan anomali tespit teknikleri açıklanmış ve sonrasında günümüzde uygulanan yüksek teknolojiye sahip derin öğrenme tabanlı anomali tespit tekniklerinin, öncesinde kullanılan geleneksel algoritmaların sorunlarını aşma konusunda kullandıkları teknikleri tartışmışlardır [197].

Hranisavljevic ve ekibi tarafından Derin Ağ Zamanlı Otomat (DENTA) isimli bir model önerilmiş ve test edilmiştir. Sonucunda özellikle gerçek dünya veri kümeleri üzerinde anomali tespiti konusunda avantajlı sonuçların elde edildiği bildirilmiştir [198].

Kim ve ekibi tarafından önerilen Projeksiyon Yolu Boyunca Yeniden Yapılanma (RaPP) isimli yenilik tespit sistemi modeli kapsamında oluşturdukları otomatik kod çözücülerin (AE) değerlendirilmesinde ise çok çeşitli özelliklere sahip veri setleri sınanmıştır. Bu sınama sürecinde kullanılan veri setlerinden bir tanesi de yüksek raflı depolama sistemlerinin enerji optimizasyon anomali tespiti için oluşturulmuş olan veri kümesidir.

Genel anlamda önerilen modelin analizler sonucunda iyi bir performans gösterdiği ifade edilmiştir [199].

Zeki fabrikalarda üretim esnasında elde edilen verilerin analizinde, yüksek oranda anormal veri olduğunu tespit eden Kim ve arkadaşları, bu verilerinin analizini daha hızlı ve düşük bir maliyetle gerçekleştirilmesini sağlayabilmek için özel bir sanal sistem kurmuşlardır. Bu sanal sistem ile anomali tespiti için veri analizine başlamadan önce, veri üzerinde filtreleme yaparak, normal veriler ile anormal verileri birbirinden ayrıştırmak için bir çerçeve önermişlerdir. Çalışmanın sonraki süreçlerinde anomalileri tiplerine göre ayrıştırabilmek için sadece anomalileri içeren veri seti ile analiz yapılabileceği belirtilmiştir [200].

Tien ve arkadaşları, anomali tespiti konusunda, ağ trafiğinin analiz edilmesi ve veri paketlerinin özeliklerine göre makinenin tespit edilmesi yöntemi üzerine çalışmışlardır. Uyguladıkları makine öğrenmesi yöntemi ile ağda akan veriler üzerinden makine tespitini XGBoost algoritması tarafından yüksek başarım oranı ile gerçekleştirmişlerdir. Çalışma sonucunda doğru makine tespiti testlerinde %97,6 doğruluk oranına erişmişlerdir. Çalışmanın ilerleyen safhalarında akış verilerinin gerçek zamanlı test edilmesi ile oluşabilecek herhangi bir anomalinin tespit edilmesi sağlanmıştır. Ancak yapılan anomali tespiti sadece anomali var veya yok şeklinde gerçekleşmektedir. Anomalinin nedeni ve tip bilgisi belirlenememektedir. Ek olarak bu çalışmanın tüm IoT cihazlarına genellenebilmesi için daha kapsamlı çalışmalar yapılması gerektiği yazarlar tarafından belirtilmiştir. [201].

Farklı bir çalışmada, zeki fabrikaların artan önemi doğrultusunda karşılaşabilecekleri siber saldırıların, anomali tespit yöntemleri ile tespit edilmesini ve hızlı bir şekilde önlem alınmasını sağlayabilecek birleştirilmiş öğrenme modeli önerilmiştir. İzleme ve algılama süreçlerini, genele yaymaktan ziyade, küçük yerel bölgelere dağıtarak çalışmasını sağlayan model, bu sayede düşük kaynak tüketimi gerçekleştirerek başarılı bir şekilde siber saldırı tespiti ve önlem alma işlemlerini gerçekleştirebilmiştir [202].

Bir başka çalışmada, zeki lojistik kullanımına uyarlanmış olan hücresel IoT sistemleri için bir anomali tespit çerçevesi önerilmiştir. Gerçek dünya verileri ile oluşturulan sistemde, anomali tespit sistemini IoT ağı içerisindeki konumuna göre yanıt verme hızı ve doğruluk oranları karşılaştırılmıştır. Önerilen çerçeve otomatik kodlayıcıya dayalı anomali tespit

modüllerini hem yerel olarak cihazlarda hem de IoT ağında çalıştırılmıştır. Sonuç olarak otomatik kodlayıcıların, makine öğrenmesi algoritmaları ile anomali tespit sistemlerinde kullanılmasının faydalı olduğunu tespit edilmiştir. Test süreçlerinde konum tabanlı ve davranışa dayalı anomaliler olmak üzere iki türün tespiti yapılabildiği ifade edilmiştir. Davranışa dayalı anomalilerin, verilerin üretilmesinde kullanılan kabın kasıtlı olarak sarsılmasıyla oluşturdukları belirtilmiştir. Ayrıca veri seti içerisinde konum tabanlı anomali verisi bulunmadığından da çalışmada bahsedilmiştir [203].

Stehmann ve Rieger ise yaptıkları çalışmada, endüstri 4.0 kapsamında anomali tespit sistemleri için teorik bir alt yapı oluşturmayı amaçlamışladır. Çalışma kapsamında 44 farklı yayın incelenmiştir ve anomali tespiti için üst düzey gereksinimler ile özel gereksinimler tespit edilmiştir. Çalışmanın sonucuna göre, literatürde anomali tespit işlemleri için en yaygın kullanılan gereksinimler, veri üretimi için sensörlerden faydalanılması, veri işleme süreçlerine ait alt yapıların ölçeklenebilir olması ve hızlı veri üretimi ve iletimi sağlayan iletişim ara yüzlerinin kullanılması olarak belirlenmiştir [204].

Planlanmamış arızaların tespiti için petrokimya endüstrisinde üretilen gerçek veriler üzerinde farklı bir çalışma yapılmıştır. Çalışmada, sistemin kullanılabilirliğini daha iyi seviyelere çıkarmak için derin öğrenme ve torbalı karar ağaçları ve yarı denetimli ML modellerinin kullanıldığı bir yaklaşım önerilmiştir. Çalışma sonucunda vakum sistemini bozan bir anomalinin başladığının tespit edildiği belirtilmiştir [205].

Araştırmalar sonucunda, üretim sistemlerine yönelik AutoML kütüphaneleri ile anomali tespiti konusunda yeterli çalışmaya rastlanmamıştır. Ayrıca anomali tespiti için yapılan çalışmalar çoğunlukla herhangi bir anomalinin varlığını belirlemeye yönelik gerçekleştirilmiştir. Çok türlü anomali tespiti çalışmalarının sayısı yeterli görülmemiştir. Bu çalışmada, öncelikle bir zeki üretim sistemi simülasyonu ile üretilen sentetik veriler ile uygun öğrenme modellerinin oluşturulması amaçlanmıştır. Sonrasında ise başarısı tespit edilen öğrenme modelinin kullanımı ile çalışan sistemde anlık üretilen veriler üzerinde gerçek zamanlı anomali tespiti yapan bir simülasyon sisteminin geliştirilmesi amaçlanmıştır. Ayrıca oluşturulan sistemin tespit ettiği anomalileri de tiplerine göre ayrıştırması ve buna göre bilgi vermesi istenen sonuçlar içerisindedir.

6. YÖNTEM

Bu tez kapsamında, endüstri 4.0 için bir zeki üretim hattında makine öğrenmesi ile elde edilen öğrenme modellerini kullanarak, gerçek zamanlı anomali tespiti gerçekleştirebilecek ve tespitleri bir ekran vasıtası ile denetçiye bildirebilecek bir sistem çerçevesi önerilmiştir. Çalışmada, gerçek zamanlı anomali tespit uygulaması yapılabilmesi için, gerçek üretim sistemine ait verilerin elde edilmesindeki zorluklardan dolayı sentetik verilerin kullanımını tercih ettik. Sentetik verileri oluşturmak için de ayrık olay simülasyonu yapısına dayalı bir patlamış mısır üretim sistemi simülasyonu oluşturduk. Veri setlerinin içerisinde öncelikle kural tabanlı anomalilerin sınıflandırılmasını sağladık. Elde edilen veri setlerini öğrenme ve test kümelerine ayrıştırarak modellerin oluşturulmasını sağladık. Sonrasında ise gerçek zamanlı olan dinamik veriler üzerinde oluşabilecek anomalilerden ne kadarının yakalandığını makine öğrenmesi algoritmaları ve AutoML kütüphaneleri ile ölçmeye çalıştık. Sistem içerisinde elde edilen anomaliler nokta anomali sınıfına girmektedir.

Bu tezin temel katkısı dört yönlüdür. İlk olarak, sistem içerisinde arızaya sebep olabilecek olan anomaliler çeşitlendirilmişlerdir. Devamında sistemin anomali tespiti sırasında, hangi anomali türünün gerçekleştirildiğini bulması da çalışmanın odak noktasını oluşturmaktadır. Çalışma, başarılı anomali tespit sisteminin gerçekleştirilmesi için makine öğrenmesi algoritmaları ve AutoML kütüphanelerinden destek alınarak en uygun öğrenme modelinin elde edilmesi yöntemine dayanmaktadır. Sonuç olarak kurulan yapı, bir simülasyon içerisinde elde edilen veriler üzerinde test edilmektedir. İlgili çalışmalar bölümünde yapılmış olan atıflardan da görüleceği üzere anomali tespiti konusunda yapılan araştırmalarda, çalışmaların çoğunlukla sistem içerisinde herhangi bir anomalinin var olup olmadığını tespit etmeye yönelik olduğu görülmüştür. Bu işlem için de hem statik veri setleri hem de dinamik veri yapılarının kullanıldığı çalışmalar mevcuttur. Bizim çalışmamızda ise farklı olarak, çoklu cihazda anomalinin varlığının tespit edilmesi ile beraber hangi cihazda hangi anomalinin tespit edildiği noktasında da bilgi veren bir yapı ortaya çıkartılmıştır.

Tez çalışması süresince simülasyon oluşturulduktan sonra ilk etapta veri setleri kayıt edilmiş ve bu veri setleri ile makine öğrenmesi algoritmaları ve AutoML kütüphaneleri kullanılarak oluşturulan öğrenme modellerinin başarımları karşılaştırılmışlardır.

Sonrasında ise her cihaz için uygun görülen iki öğrenme modeli kullanılarak, akış verisi üzerinde gerçek zamanlı anomali tespiti yapabilmeleri konusunda testler uygulanarak sonuçlar karşılaştırılmış ve bu sonuçlara göre simülasyonun çalışması esnasında gerçek zamanlı olarak anomali tespiti cihazında kullanılacak öğrenme modellerine karar verilmiştir. Son olarak simülasyona tercih edilen öğrenme modelleri ile anomali tespiti yapısının eklenmiş ve denetçiye tespit edilen anomalilerin bilgilerinin aktarılmasını sağlayacak düzen oluşturulmuştur.

6.1. Çalışmanın Önemi ve Amacı

Günümüz teknolojisinin geldiği nokta ile endüstri uygulamalarında birbirleri ile bağlantılı ağ yapılarının entegrasyonu ile ilgili çalışmalar artmıştır. Bu sayede üretim sistemleri daha ileri seviye teknolojiye sahip otonom bir yapıya kavuşmuştur. Otonom üretim sistemleri içerisinde kullanılan cihazların, kararlı çalışabilmeleri için birbirleri arasında sağlıklı veri aktarımı sağlayabilmelerine ihtiyaçları vardır. Cihazlar arasında hatalı veri aktarımı gerçekleştiği takdirde, bu durum ilgili işletme için büyük sorunlara yol açabilecektir. Cihazların yanlış veri üretmesi, dışarıdan yapılabilecek müdahaleler veya kullanıcı hataları sebebiyle oluşması muhtemel olumsuz senaryoların önüne geçebilmek için otonom üretim sistemlerinin tasarımında güvenlik konusuna da azami derecede önem göstermek gerekmektedir. Bu tür sistemler içerisinde alınabilecek en önemli güvenlik önlemlerinden biri de sistemin çalışması anında ortaya çıkabilecek hataların hızlı bir şekilde gerçek zamanlı olarak tespit edilebilmesi ve yetkililere hızlıca sorun bilgisinin aktarılmasıdır. Bundan dolayı anomali tespit çalışmaları, endüstri 4.0 entegrasyonları kapsamında öne çıkan güvenlik konularından biri olmuştur.

Bu çalışmada bir otonom üretim sisteminin örnek bir simülasyonu geliştirilerek, analiz için gerekli verilerin üretilmesi ve bu veriler ile yapılan çalışmalar sonucunda, üretim sistemi bünyesinde oluşabilecek bir anomalinin, nedeni ve adres bilgileri ile birlikte hızlı bir şekilde gerçek zamanlı olarak tespit edilmesi amaçlanmıştır.

6.2. Çalışmanın Kapsamı

Çalışmada, endüstri 4.0 çatısı altında değerlendirilen zeki üretim sistemleri içerisinde oluşabilecek anomalilerin tespitine odaklanılmıştır. Bu sebeple, endüstri 4.0 teknolojisine entegrasyon sağlamış veya bu konudaki girişimlere devam eden küçük ve orta büyüklükteki üretim işletmeleri, bu işletmelerin içerisinde kullanılan siber fiziksel sistemlere uygun cihazlar ve bu cihazların üretecekleri veriler çalışmanın kapsamını oluşturmaktadır.

6.3. Çalışmanın Sınırlılıkları

Bu tez çalışmasının sınırlılıkları aşağıda listelenmiştir:

- Gerçek zamanlı anomali tespit uygulaması yapılabilmesi için, gerçek üretim sistemine ait verilerin elde edilmesindeki zorluklardan dolayı sentetik verilerin kullanımını tercih edilmiştir.
- Sistemin çalışması süresince oluşması muhtemel anomali türleri, önceden belirlenmiş ve veri setleri içerisine sınıf etiketi olarak kodlanmıştır.
- Tez çalışması kapsamında sınıflama tabanlı anomali tespit yaklaşımı benimsendiği için, önceden belirlenmiş anomali türlerinin tespitine yönelik bir çalışma ortaya çıkartılmıştır.

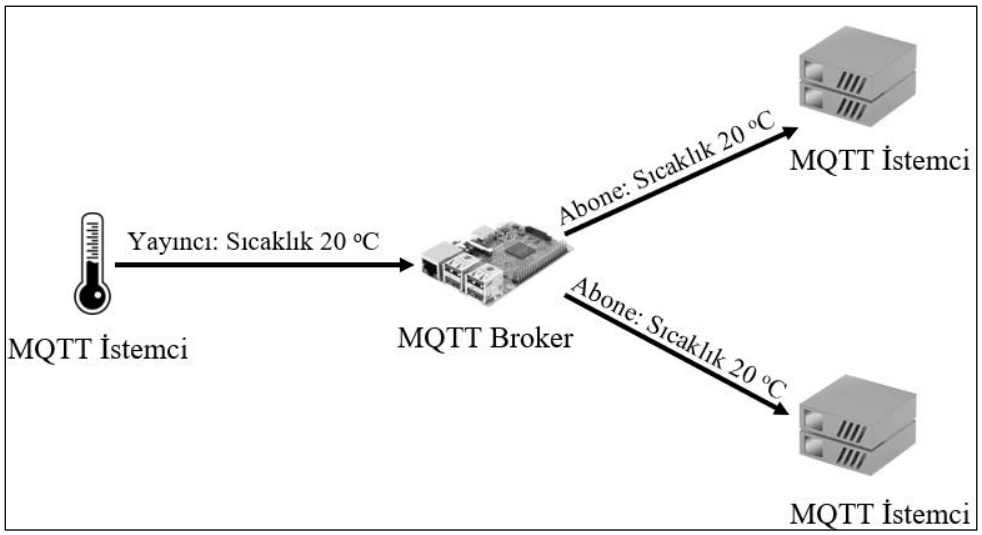
7. ANOMALİ TESPİT SİSTEMİ GELİŞTİRİLMESİ

Bu bölümde tez kapsamında geliştirilen anomali tespit sisteminin analiz, tasarım, geliştirme ve değerlendirme süreçlerine ait ayrıntılı bilgileri verilmiştir.

7.1. Analiz

Günümüzde, M2M cihazlarında iletişimin gerçekleşmesi amacıyla birçok protokol kullanılmaktadır. Kullanılan protokoller içerisinden en önemli 5 protokol; Hiper Metin Aktarım Protokolü (Hyper Text Transfer Protocol - HTTP), Genişletilebilir Mesajlaşma ve Durum Protokolü (Extensible Messaging and Presence Protocol - XMPP), Gelişmiş Mesaj Kuyruklama Protkolü (Advanced Message Queuing Protocol – AMQP), Sınırlı Uygulama Protokolü (Constrained Application Protocol - CoAP) ve MQ Telemetri Taşıma (MQ Temeletry Transport - MQTT) şeklinde sayılabilir. Makineler arası veri aktarımı için kullanılacak protokol seçiminde, enerji verimliliği, performans, kaynak kullanımı ve güvenilirlik kavramlarını göz önünde bulundurmak gerekmektedir. Bu hususlar baz alındığında, yukarıda sayılmış olan protokoller içerisinden MQTT protokolü, en iyi seçeneklerden biri olacaktır [206].

Çalışma kapsamında, siber fiziksel sistemler içerisinde zeki çözüm platformu olarak kullanılan MQTT tabanlı veri akışı gerçekleştiren bir otonom üretim sistemi simülasyonu ve bu simülasyonu hatalara karşı daimi olarak kontrol eden bir anomali algılama sistemi oluşturulmuştur. Oluşturulan yapıda, ERP tarafından yönetilen üretim sistemi cihazları bulunan, ERP ile diğer cihazlar arasında ve cihazların kendi aralarında MQTT Broker aracılığı ile veri alışverişi gerçekleştirmeleri sağlanmıştır.



Şekil 7.1. MQTT sistem yapısı [207]

MQTT, yayın / abonelik sistemine dayanan, basit ve düşük kaynak tüketimine sahip bir mesajlaşma protokolüdür (Şekil 7.1). Genel anlamda kısıtlı cihazlar, düşük bant genişliği ve yüksek gecikme süresi / güvenilmez ağlar için tasarlanmıştır. Tasarım ilkelerine göre bu protokol, ağ bant genişliği kullanımı ve cihaz kaynak kullanımını en aza düşürürken, bir seviyeye kadar mesajın güvenliğini sağlamaya çalışmaktadır. Bu özellikleri sayesinde MQTT protokolü, M2M ve nesnelerin interneti gibi ortamlarda kullanılabilecek ideal bir protokol olarak gösterilmektedir [207].

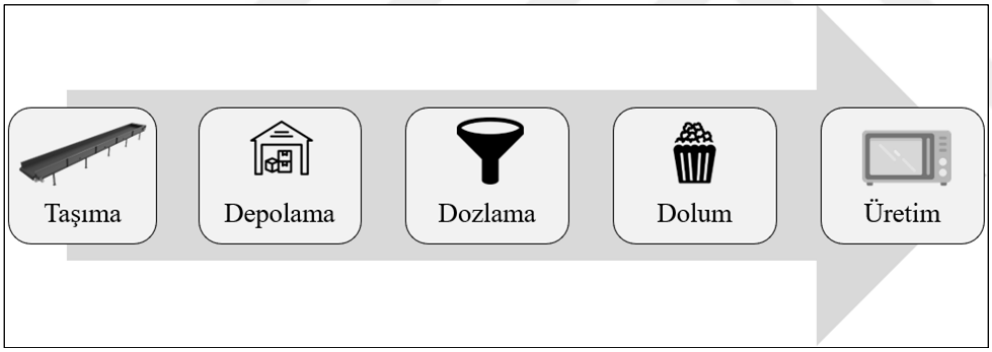
MQTT, istemci cihazların birdenbire güncelleme talep etme ihtiyacı duymadan, yayın / abonelik mantığını uyguladıkları için mesaj odaklı bir iletişim sistemidir. MQTT, "Broker" olarak adlandırılan bir merkez ve bu merkeze bağlı cihazlar ile sunucu-istemci mimarisini kullanmaktadır. Bu sistem içerisinde cihazlar ve ERP, direkt olarak birbirleri ile haberleşmezler. Tüm cihazlar Broker'a bağlanır ve onun üzerinden iletişim kurarlar. MQTT içerisinde mesajlar konu başlıklarına ayrılırlar ve cihazlar kendileri ile ilgili olan konulara abone olurlar. Broker, yayıncı konumunda olan cihazdan gelen mesajı alır ve mesajın ait olduğu konuya abone olan cihazlara iletimini gerçekleştirir. Bu şekilde cihazlar arasında, 1-n veya n-m ilişki türünde asenkron bir iletişim hattı oluşturulmuş olur [82].

Uygulama simülasyonu olarak bir patlamış mısır üretim sistemi barındıran zeki fabrika modülü oluşturulmuştur. Simülasyon kapsamında beş adet cihaz, bir adet ERP, bir adet anomali tespit merkezi ve bir adet anomali takip ekranı örneklenmiş ve gerekli veri setlerinin oluşturulması sağlanmıştır.

Simülasyonların oluşturulması için Intel Core i7 9750H 2.6 Ghz CPU, 6 Gb RAM ve 3 Gb Nvidia GeForce GTX 1050 ekran kartı donanımına ve Ubuntu 20.0 işletim sistemine sahip bir bilgisayar kullanılmıştır. Bilgisayara kurulmuş olan Docker yazılımı ile her bir fiziksel cihazı temsil etmesi için ilgili konteynerler oluşturulmuş ve bu konteynerlerin içerisine, Python programlama dili ile yazılmış olan cihaz simülasyonları kaydedilmiştir.

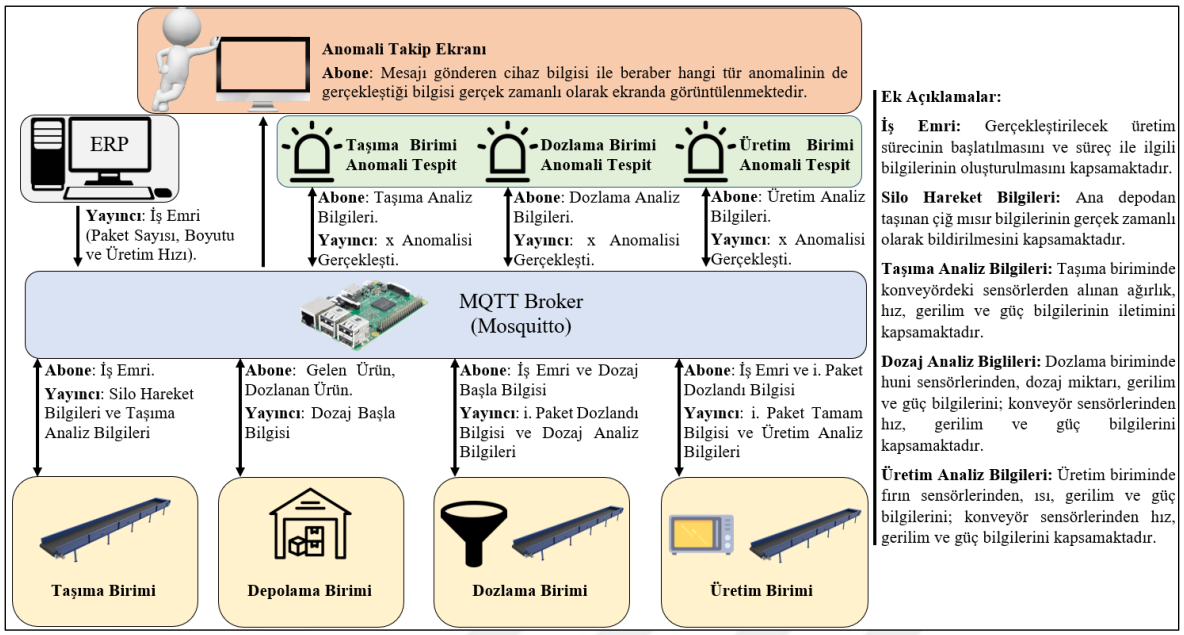
7.2. Tasarım ve Geliştirme (Patlamış Mısır Üretim Sistemi Simülasyonu)

Bu simülasyonun oluşturulması amacıyla, kaggle ortamından çekilmiş olan veri setleri örnek alınmıştır [208]. Örnek alınan sistem, taşıma, depolama, dozlama, dolum ve üretim olmak üzere 5 modülden oluşacak şekilde geliştirilmiştir (Şekil 7.2). Sistem içerisinde bulunan her modüle ait olan birimlerin gerçekleştirmiş olduğu işlemler kayıt altına alınarak, her modül için ayrı ayrı veri setleri oluşturulmuştur.



Şekil 7.2. Patlamış mısır üretim sistemi iş akış sırası [208]

Simülasyon olarak hazırlanan sistemde ise patlamış mısır üretim sistemine, iş emri doğrultusunda gereken çiğ mısır ürününün 5000 gr ağırlığında partiler ile ara depolama birimine taşınması, sonrasında buradan gerekli miktarda dozlama yapılarak üretim birimine aktarılmasının simule edilmesi üzerine kurulmuştur. Ubuntu işletim sistemi üzerinde, docker sanallaştırma sistemi ile konteynırlar oluşturulmuş ve python programlama dili ile cihazların simülasyonları yazılarak konteynırlara aktarılmıştır. Bu sayede her bir konteynırın üretim sisteminin bir parçasıymış gibi çalışarak veri üretmesi hedeflenmiştir. Kurulan simülasyon, Kurumsal Kaynak Planlama (ERP) Cihazı, Taşıma Birimi, Depolama Birimi, Dozlama Birimi ve Üretim Biriminden oluşmaktadır (Şekil 7.3). Simülasyon, Ayrık Olay Simülasyonu yapısına göre oluşturulmuştur.



Şekil 7.3. Simülasyon mimarisi

Oluşturulan simülasyonda, Taşıma, Dozlama ve Üretim her biri için ayrı ayrı veri setleri anomali tespit işlemlerinde kullanılmak üzere üretilmiştir. Birimler için ayrı veri setleri, her birimin kendi içerisinde bulunan çalışma sistemine özgü anomali tespit sürecinin gerçekleştirilmesi amaçlanmaktadır.

Ayrıca ERP tarafından verilen iş emrindeki bilgiler, süreç boyunca diğer birimler tarafından da takip edilmektedir. Bu sayede, iş emrinde belirtilenin aksine bir işleyiş gerçekleştiğinde sistem bunu tespit ederek, anomali takip ekranına aktarması planlanmaktadır. Simülasyonda oluşturulan birimler ile ilgili ayrıntılı bilgiler alt başlıklarda aktarılmıştır.

7.2.1. Kurumsal kaynak planlama (Enterprise resources planning – ERP)

ERP birimi sistemin üretim süreçlerinin yönetimi ve iş emirlerinin oluşturulması amacıyla kurulmuştur. Sistem 3 farklı çeşitte üretim kapasitesine sahiptir. Bu çeşitler küçük boy paket (100g), orta boy paket (250g) ve büyük boy paket (500g) olacak şekilde belirlenmiştir. ERP uygulaması çalıştırıldığında, ana ekranda kullanıma hazır mevcut çiğ mısır miktarı ve stokta hangi çeşit kaç paket mevcut olduğu bilgileri verilmektedir. Ayrıca sistem 3 farklı hız seçeneği ile üretim gerçekleştirmektedir. Uygulanacak üretim sürecinin zaman ihtiyacına göre istenilen hız seçeneği seçilebilmekte ve buna göre sürecin gerçekleşmesi ve tamamlanması sağlanmaktadır.

ERP modülü, iki farklı kullanım seçeneğine sahiptir. Birinci seçenekte direk stok üzerinden seçilen paket türünde satış bilgilerinin girilmesine ve satılan ürün adedinin stoktan düşülmesine imkan sağlamaktadır. Bu yapıda, eğer satış sonunda ürün miktarı 100 adedin altına düşerse, otomatik olarak üretim sistemi devreye girerek ilgili paketten 250 adet üretim gerçekleştirilmesi sağlanmaktadır. İkinci uygulamada ise, kullanıcı girişi yapılarak istenilen çeşit üründen, yine istenilen adet kadar stok için üretim yapılması sağlanmaktadır.

ERP modülü üzerinden üretim emri girilmesi veya satış emri girişi yapıldıktan sonra stok miktarının belirtilen sınırın altına düşmesi durumunda sistem Taşıma biriminin harekete geçmesi ile üretim süreçlerini başlatmaktadır.

7.2.2. Taşıma birimi

Taşıma birimi araç olarak, çiğ mısırın doldurulduğu bir adet silo, taşıma esnasında hava akımının düzenlenmesi amacıyla bir adet aspiratör ve dolu siloyu Depolama birimine götürecek ve boş siloyu geri getirecek olan bir adet konveyör sisteminden oluşmaktadır.

ERP den gelen bilgiler doğrultusunda Taşıma birimi, üretilecek ürün çeşidi ve adedi bilgilerini alarak gereken toplam çiğ mısır miktarını Depolama birimine taşımaya başlayacaktır. Gerekli toplam çiğ mısır miktarının hesaplanması konusunda aşağıdaki Eş. 7.1 üzerinden işlem yapılmaktadır.

$$Ham_{Urun} = \frac{(Islenmis_{Urun} * 100)}{92} + (Paket_{Sayisi} * Carpan) \quad (7.1)$$

Yukarıdaki eşitlikte bulunan değişkenler;

- $\underline{Ham_{Urun}}$: gerekli olan çiğ mısır ağırlığını,
- $\underline{Islenmis_{Urun}}$: Üretim sonunda elde edilmesi gereken toplam işlenmiş mısır ağırlığını,
- $\underline{Paket_{Sayisi}}$: Üretilmesi gereken toplam paket sayısını,

- Carpan: Üretilecek ürün çeşidi ve paket sayısına bağlı olarak, ihtiyaç duyulması ihtimaline karşın yedek çiğ mısır miktarının toplam ağırlığa eklenmesini sağlayacak olan değeri (Küçük Boy için Carpan= 1; Orta Boy için Carpan= 2; Büyük Boy için Carpan= 3) temsil etmektedirler.

Üretim için gerekli olan çiğ mısır miktarı hesaplandıktan sonra Taşıma birimi her seferde ortalama 5000g (4850g – 5150g) olacak şekilde çiğ mısırları Depolama birimine taşımaya başlayacaktır. Taşıma birimi her bir seferini, 400 adımda ve tercih edilen hız seçeneğine göre 30, 35 veya 40 saniyede gerçekleştirmektedir. Adımlar arasındaki süre ortalama 0,079 sn ila 0,108 sn olmaktadır. Her adımda, Unix zaman damgası, silo doluluk bilgileri, aspiratör ve konveyörün çalıştıklarını ve durduklarını gösteren bilgileri, konveyörün taşıma esnasında aldığı mesafe, harcadığı güç ve voltaj bilgileri, anlık taşınan çiğ mısır miktarı, o ana taşınmış olan çiğ mısır miktarı ve taşınması gereken kalan mısır miktarına ait bilgileri üretmektedir. Bu bilgiler aynı zamanda veri tabanına kaydedilmekte ve ihtiyaç duyulanlar ilgili diğer makinelere iletilmektedir.

Taşıma birimine özgü anomali tespit sisteminin kullanılabilmesi amacıyla oluşturulan veri setine ait alan bilgileri Çizelge 7.1'de görülmektedir.

Çizelge 7.1. Taşıma birimi veri seti nitelik tanımları

Nitelik Adı	**Açıklaması**
ZamanID	Sistem Çalışma Hızı Bilgisi (Nümerik)
SiloFull	Silonun Tam Doluluk Bilgisi (0 – 1)
SiloMinFull	Silonun Minimum Seviye Doluluk Bilgisi (0 – 1)
Asprator	Aspiratörün Çalışma Durumu Bilgisi (0 – 1)
Konveyör	Konveyörün Çalışma Durumu Bilgisi (0 – 1)
Mesafe	Konveyör Aldığı Mesafe Bilgisi (Nümerik)
Guc	Konveyörün Anlık Harcadığı Güç Bilgisi (Nümerik)
Voltaj	Konveyörün Anlık Kullandığı Gerilim Bilgisi (Nümerik)
Aktif	Anlık Taşınan Çiğ Mısır Miktarı (Nümerik)
Sinif	Anomali Durum Bilgisi (0 – 1 – 2 – 3 – 4)

Çizelge 7.1'de gösterilen değişkenlerden Mesafe bilgisi, konveyörün her adımda 30 birim hareket etmesi ile elde edilen değerdir. Silonun dolu bir şekilde depolama birimine taşınması esnasında mesafe değeri artan şekilde hesaplanmaktadır; boş silonun geri getirilmesi esnasında ise, azalan şekilde hesaplama süreci işlemektedir. Voltaj değerinde, şebekeden gelen 220V ile 230V arasındaki anlık ölçülen değer gösterilmektedir.

Konveyörün harcadığı güç değerinin hesaplanması için ise Eş. 7.2 ve Eş. 7.3'te gösterilen Konveyör Güç Eşitlikleri kullanılmıştır.

$$P = \frac{A * m * v}{1000 * \eta} \tag{7.2}$$

$$A = \left(\frac{2}{D} * \left(\mu * \left(\frac{d}{2}\right) + f\right) + sin\,\alpha\right) * g \tag{7.3}$$

Konveyör güç hesaplama işlemi içi kullanılan yukarıdaki eşitliklerde belirtilen değişkenlerin anlamlar şu şekildedir [209]:

- P: Konveyör tarafından harcanan güç değeri (W)
- m: Bant üzerindeki toplam bant + yük ağırlığı (kg)
- v: Bant hızı (m/sn)
- η: Verim
- D: Destek tamburları dış çapı (mm)
- d: Destek tamburları göbek çapı (mm)
- f: Yuvarlanma direnci
- α: Konveyörün yatay açısı
- g: Yer çekimi ivmesi (m/sn^2)
- μ: Sürtünme direnç katsayısı

7.2.3. Depolama birimi

Depolama birimi, Taşıma ve Dozlama birimleri arasında aktarma olarak görev yapmaktadır. Taşıma birimi tarafından her seferde gelen ortalama 5000g ağırlığındaki çiğ mısır depoya alınır. İlk parti çiğ mısır depoya alındığı andan itibaren Dozlama birimine sinyal ulaşır. Sinyalin gelmesi ile birlikte otomatik olarak çalışmaya başlayan Dozlama birimi, her bir paketin üretimi için gerekli olan çiğ mısır miktarını Depolama biriminden çekecektir.

Birim, kendisine Taşıma biriminden gelen ve Dozlama birimi tarafından kendisinden çekilen çiğ mısır miktarlarını daimi olarak kayıt altına almaktadır. İlgili üretim serisinde,

son paket için gerekli olan çiğ mısır miktarı da Dozlama birimi tarafından çekildikten sonra birim, Taşıma birimine sinyal göndererek, kalan çiğ mısırların Taşıma birimi aracılığı ile hammadde stok deposuna aktarılmasını sağlamaktadır. Ayrıca Dozlama birimi tarafından yapılan dozlama işlemlerinde, gereğinden fazla miktarda çiğ mısır dozlaması yapılması sonucunda, Depolama biriminde stoklanan çiğ mısır miktarı, toplam üretim miktarını gerçekleştirmeye yetmeyebilir. Bu durumda Depolama birimi, üretim süreçlerine ara verilmesini ve gerekli ek çiğ mısır miktarının tamamlanması için Taşıma biriminin tekrar çalışmaya başlamasını sağlamaktadır. Ek olarak yaşanan bu olumsuz durum, anomali takip ekranına yansıtılarak, denetçilerin bilgilendirilmesi sağlanmaktadır. Üretim süreçleri haricinde Depolama birimi içerisinde çiğ mısır stoku yapılmamaktadır.

7.2.4. Dozlama birimi

Dozlama birimi içerisinde araç olarak, uygun miktar dozlamanın yapılması amacıyla bir adet huni, depodan çekilen mısırın üretim birimine taşınması için bir adet konveyör ve cihaz içerisindeki hava akımının dengelenebilmesi amacıyla bir adet aspiratör bulunmaktadır.

Taşıma birimi tarafından ilk parti çiğ mısırın, Depolama birimine aktarıldığına dair sinyal ulaştığı andan itibaren beş saniye sonra Dozlama birimi çalışmaya başlamaktadır. Ürün paket tipine göre gerekli olan çiğ mısır miktarı hesaplanmakta ve huni tarafından hesaplanan miktar Depolama biriminden çekilmektedir. Her paket için huni ile dozlanacak çiğ mısır miktarı hesaplanması için Eş. 7.4 kullanılmaktadır.

$$Doz_{Miktarı} = \frac{Paket_{Boyutu} * 100}{Random\ (91.5 : 92.5)} \tag{7.4}$$

Eş. 7.4'te belirtilen;

- $Doz_{Miktarı}$**:** İşlem sonucunda ortaya çıkan ve huni tarafından dozlanacak çiğ mısır miktarını,
- $Paket_{Boyutu}$**:** Üretim sonucunda ortaya çıkacak olan ürünün ağırlığını,
- Random**:** Uygun çiğ mısır miktarının hesaplanabilmesi amacıyla 91,5 – 92,5 arasında rastgele üretilen değeri temsil etmektedirler.

Eş. 7.4 sonucunda, üretim sonucunda elde edilmesi gereken paket ağırlığının %108,1 ile %109,3 oranında çiğ mısırın dozlanması sağlanmaktadır. Dozlama işlemi gerçekleştirilen çiğ mısır, sonrasında konveyör aracılığı ile Üretim birimine taşınmaktadır.

Bu birimde huni ile dozlama ve konveyör ile çiğ mısırın taşınması işlemleri toplamda 400 adımda gerçekleşmektedir. Süreçlerin gerçekleşme süresi ise tercih edilen üretim hızı değeri ve dozlama işleminde çekilen çiğ mısırın ağırlığına göre değişmekte ve sistem bu durumu kendisi ayarlamaktadır. Hız tercihi ve üretilecek paket türlerine göre Dozlama biriminin bir paket için gerekli tüm işlemler boyunca harcadığı süre değerleri;

- Normal üretim hızı tercihi ile,
    - 100 gr paket için 36 sn.,
    - 250 gr paket için 37,7 sn.,
    - 500 gr paket için 40 sn.'dir.
- Hızlı üretim hızı tercihi ile,
    - 100 gr paket için 31,5 sn.,
    - 250 gr paket için 33 sn.,
    - 500 gr paket için 35 sn.'dir.
- Çok hızlı üretim tercihi ile,
    - 100 gr paket için 27 sn.,
    - 250 gr paket için 28,3 sn.,
    - 500 gr paket için 30 sn.'dir.

Dozlama biriminin kendine özgü anomali tespit süreçlerinde kullanılmak üzere oluşturulan veri setine ait alan bilgileri Çizelge 7.2'de verilmiştir.

Çizelge 7.2. Dozlama birimi veri seti nitelik tanımları

Nitelik Adı	**Açıklaması**
Hiz	Sistem Çalışma Hızı Değeri
ZamanID	Çalışma Zamanı Bilgisi
PaketBoyutu	O An Üretilen Ürünün Paket Tipi (Nümerik)
DozMiktarı	Üretim İçin Dozlama İşlemi Yapılan (Nümerik)
Asprator	Aspiratörün Çalışma Durumu Bilgisi (0 – 1)
Huni	Huninin Çalışma Durumu Bilgisi (0 – 1)
HuniKapasite	Huni Tarafından Anlık Çekilmiş Olan Çiğ Mısır Miktarı (Nümerik)
HuniGuc	Huni Tarafından Anlık Harcanan Güç Bilgisi (Nümerik)
HuniVoltaj	Huni Tarafından Anlık Kullanılan Gerilim Bilgisi (Nümerik)
Konveyör	Konveyörün Çalışma Durumu Bilgisi (0 – 1)
KonvMesafe	Konveyörün Aldığı Mesafe Bilgisi (Nümerik)
KonvGuc	Konveyörün Anlık Harcadığı Güç Bilgisi (Nümerik)
KonvVoltaj	Konveyörün Anlık Kullandığı Gerilim Bilgisi (Nümerik)
Sinif	Anomali durum bilgisi (0 – 1 – 2 – 3 – 4 – 5 – 6)

Çizelge 7.2'de ifade edilen değişkenlerden huni ve konveyör voltaj değerleri, şebekeden anlık olarak gelen 220V ile 230V arasındaki veriyi göstermektedir. Huni tarafından harcanan gücün hesaplanmasında, dozaj motoru güç eşitliği örnek alınarak Eş. 7.5 oluşturulmuştur [210]. Konveyörün güç hesabında ise, Taşıma biriminde de uygulanan Eş. 7.2 ve Eş. 7.3'te gösterilen konveyör güç hesabı eşitlikleri kullanılmıştır.

$$P = \frac{Doz * h * hiz}{367 * n} \tag{7.5}$$

Eş. 7.5'te belirtilen değişkenlerin anlamlar şu şekildedir:

- P: Huninin çalışması esnasında harcadığı anlık güç değeri (W),
- Doz: Dozlama yapılan çiğ mısır miktarı (gr),
- h: Huni yükseklik bilgisi (1 olarak belirlenmiştir),
- hiz: Sistem üretim hızı değeri (m/sn),
- n: Pompa tesir derecesi (0,5 olarak belirlenmiştir),
- 367: Eşitlik sabitidir.

7.2.5. Üretim birimi

Üretim birimi içerisinde araç olarak, çiğ mısırın patlatılması amacıyla bir adet fırın, bir adet paketleme aracı, paketlenmiş olan ürünün depoya taşınması için bir konveyör ve cihaz içerisindeki hava akımının dengelenmesi için bir adet aspiratör bulunmaktadır.

Dozlama birimi tarafından getirilen çiğ mısır, Üretim birimi içerisinde fırına aktarılmaktadır. Sıcaklığı ortalama 200 ^{o}C (195 – 205) sıcaklıkta olan fırın içerisinde üretilen patlamış mısır, üretim işlemi tamamlandıktan sonra paketleme sistemine alınmaktadır. Buradaki işlemden sonra ise paketlenmiş ürün, konveyör yardımı ile stoka taşınmaktadır. Üretim birim kapsamında bir paket ürünün fırında işlenmesi ve sonrasında konveyör ile taşınması süreçleri toplamda 400 adımda gerçekleşmektedir. Sistem her adımda Unix zaman damgası ile birlikte Çizelge 7.3'te gösterilen nitelik değerlerine ait verilerin kayıtlarını gerçekleştirmektedir. Bir paket için gerekli işlem süresi ise ürün ağırlığına göre değişkenlik göstermektedir. Sistem hızını bu konuda kendisi ayarlamaktadır. Üretilen paket türlerine göre Üretim biriminde bir paket için gerekli tüm işlemler boyunca harcadığı süre değerleri;

- Normal üretim hızı tercihi ile,
    - 100 gr paket için 35 sn.,
    - 250 gr paket için 37,2 sn.,
    - 500 gr paket için 40 sn.'dir.
- Hızlı üretim hızı tercihi ile,
    - 100 gr paket için 30,6 sn.,
    - 250 gr paket için 32,6 sn.,
    - 500 gr paket için 35 sn.'dir.
- Çok hızlı üretim hızı tercihi ile,
    - 100 gr paket için 26,2 sn.,
    - 250 gr paket için 28 sn.,
    - 500 gr paket için 30 sn.'dir.

Normal şartlarda çiğ mısır ile üretim sonucu elde edilen patlamış mısır miktarlarının farklılık göstermesi durumundan dolayı, Üretim birimi simülasyonu içerisinde, gelen çiğ mısır miktarına göre, patlamış mısır paket ağırlığının hesaplanması için Eş. 7.6

kullanılmaktadır. Verilen eşitliğe göre, Üretim birimine gelen çiğ mısır miktarının %91,5 ile %92,5 oranında patlamış mısır elde edilmesi sağlanmaktadır.

$$Paket = \frac{Mısır_{Çiğ} * Random\ (91{,}5{:}92{,}5)}{100} \tag{7.6}$$

Eş. 7.6'da belirtilen değerler;

- Paket**:** Üretim sonucu elde edilecek paketin ağırlığını,
- $Mısır_{Çiğ}$**:** Üretim için Dozlama biriminden gelen çiğ mısır ağırlığını,
- Random**:** Simülasyon içerisinde üretilmiş patlamış mısır paketinin ağırlığının hesaplanabilmesi amacıyla 91,5 – 92,5 arasında rastgele üretilen değeri temsil etmektedirler.

Üretim biriminin kendine özgü anomali tespit süreçlerinde kullanılmak üzere oluşturulan veri setine ait alan bilgileri Çizelge 7.3'te verilmiştir.

Çizelge 7.3. Üretim birimi veri seti nitelik tanımları

Nitelik Adı	Nitelik Açıklaması
Hiz	Sistem Çalışma Hızı Değeri
PaketTip	Üretilen Paket Tipi Bilgisi (1, 2, 3)
HamUrun	Dozlama Biriminden Gelen çiğ mısır ağırlığı (Nümerik)
SonUrun	Üretim sonucundan elde edilen patlamış mısır ağırlığı (Nümerik)
Asprator	Aspiratörün Çalışma Durumu Bilgisi (0, 1)
Firin	Fırının Çalışma Durumu Bilgisi (0, 1)
FirinGuc	Fırın Tarafından Harcanan Güç Bilgisi (Nümerik)
FirinVoltaj	Fırın Tarafından Kullanılan Gerilim Bilgisi (Nümerik)
FirinSicaklik	Fırının Sıcaklık Bilgisi (Nümerik)
Konveyör	Konveyörün Çalışma Durumu Bilgisi (0, 1)
KonvMesafe	Konveyörün Aldığı Mesafe Bilgisi (Nümerik)
KonvGuc	Konveyörün Anlık Harcadığı Güç Bilgisi (Nümerik)
KonvVoltaj	Konveyörün Anlık Kullandığı Gerilim Bilgisi (Nümerik)
Sinif	Anomali Durum Bilgisi (0, 1, 2, 3, 4, 5)

Üretim biriminde kullanılan fırının gerilim değeri 380V ile 400V arasında anlık olarak ölçülen veri şeklinde, konveyörün gerilim değeri ise 220V ile 230V arasında anlık olarak ölçülen veri şeklinde kaydedilmektedir. Fırının çalıştığı esnada kullandığı güç bilgisinin hesaplanmasında ise, Eş. 7.7'de ifade edilen akım ve gerilim ilişkisi ile elde edilen güç hesaplama eşitliği kullanılmıştır. Konveyörün harcadığı gücün hesaplanmasında ise

Taşıma biriminde ve Dozlama biriminde olduğu gibi Eş. 7.2 ve Eş. 7.3'te açıklanmış olan konveyör güç hesaplama eşitliklerinden yararlanılmıştır.

$$P = V * I \tag{7.7}$$

Eş. 7.7'de belirtilen değişkenlerin anlamları şu şekildedir:

- P: Fırının çalışması esnasında harcadığı güç miktarı (W),
- V: Fırının çalışması için şebekeden gelen gerilim değeri (V),
- I: Fırının anlık olarak çektiği akım miktarıdır (A).
    - Akım değeri, normal üretim hızı tercihinde 18A, hızlı üretim hızı tercihinde 21,06A, çok hızlı üretim hızı tercihinde ise 23,94A olarak ortalama değerlerde ölçülmektedir.

7.3. Değerlendirme İşlemleri

Çeşitli çalışmalar kapsamında kullanılmak üzere elde edilen veri setlerinde, bazı verilerde eksiklikler, hatalar, tekrarlar veya anlamsızlıklar bulunabilmektedir. Bundan dolayı, veri setleri üzerinde çalışma yapmaya başlamadan önce, çeşitli önişleme süreçlerinden geçirmek önem arz etmektedir. Bu süreçler, kayıp verilerin düzenlenmesi, gürültünün ortadan kaldırılması, bütünleştirme, dönüştürme ve azaltma şeklinde isimlendirilebilmektedir [132]. Dolayısı ile araştırmacının elindeki veri setinin ihtiyaçları doğrultusunda, anılan düzenleme işlemlerinden uygun olanları değerlendirmesi gerekmektedir. Bu kapsamda, çalışma içerisinde üretilen veri setlerinin analiz süreçlerinde hatalı sonuçlar çıkmasına sebep olacak bazı nitelikler analiz aşamasında veri setlerinden çıkartılmışlardır.

Veri setleri üzerinde analiz işlemlerine başlamadan önce veri seti nitelikleri içerisinden Sınıf niteliği ile diğer niteliklerin ayrıştırılması işlemi yapılmıştır. Bu işlem, algoritmaların Sınıf niteliği haricinde kalan diğer nitelikleri kullanarak öğrenme sürecinin gerçekleştirmeleri ve öğrenme işleminde elde edilen sonucun Sınıf niteliğindeki gerçek değer ile kontrol edilerek, hata düzeltme işlemlerinin yapılmasını sağlamaktadır. Algoritmalardaki hata düzeltme işlemleri ise en iyi sonucu veren ağırlıklara ve

değişkenlere sahip modellerin oluşturulmasını amaçlamaktadır. Ayrıca, algoritmalar ile analiz öncesinde diğer niteliklerin sahip olduğu değerler üzerinde, standardizayon, normalleştirme ve min-max ölçeklendirme gibi önişleme adımları da yapılabilmektedir. Bu tür önişleme adımları veri setleri üzerinde elde edilebilecek olan başarım oranlarını doğrudan etkileyebilmektedir. Tez kapsamında oluşturulmuş olan Taşıma birimi, Dozlama birimi ve Üretim birimi veri setleri üzerinde bahsedilen önişleme süreçleri denenmiştir. Ancak veri setlerinin Sınıf etiketlerindeki değerlerin dağılımında yeterli balans sağlanamamıştır. Bu sebeple kullanılabilecek olan önişleme adımları, veri setlerinin doğruluk oranlarında yukarı yönlü bir fayda sağlamamıştır. Bundan dolayı veri setlerinin analizine başlamadan önce, veriler üzerinde önişleme adımları çalıştırılmamıştır.

Çalışma kapsamında kullanılan veri setleri üzerinde, en uygun öğrenme modellerinin tespit edilebilmesi için, makine öğrenmesi algoritmalarından Naive Bayes, Yapay Sinir Ağları, Karar Ağacı, Rastgele Orman, Lojistik Regresyon, k – En Yakın Komşu ve Destek Vektör Makineleri algoritmaları standart parametre ayarlarında bırakılarak kullanılmıştır. Ayrıca optimum hiper parametre değerlerinin daha net ortaya çıkartılabilmesi için AutoML kütüphanelerinden Auto-Keras, Auto-Sklearn ve PyCaret kütüphaneleri ile değerlendirme işlemleri yapılmıştır.

Çalışma kapsamında kullanılan veri setleri üzerinde makine öğrenmesi algoritmalarının sınanması sürecinde aşağıda belirtilen ölçüm birimleri kullanılmıştır. Bu ölçümlerden elde edilen sonuçlar doğrultusunda, veri setleri üzerinde en uygun değerlemeyi yapan öğrenme algoritmasına karar verilebilmektedir.

7.3.1. Karmaşıklık matrisi

Karmaşıklık matrisi, öğrenme modellerinin performanslarının görselleştirildiği bir düzen olarak kullanılmaktadır. Herhangi bir algoritma ile sınıflandırma işlemi yapabilmek için oluşturulmuş olan öğrenme modellerinin, gerçek sınıf değerleri bilinen test verileri üzerinde sınanmaları sonucunda, modelin başarımını gösterir. Karmaşıklık matrisi tarafından model performansının belirlenmesi amacıyla yapılan tanımlamalar şu şekildedir [105]:

- True Positive (TP): Gerçek Pozitif - Gerçekte 1 olan sınıf etiketlerinin, 1 olarak tahmin edilme sayısı,
- True Negative (TN): Gerçek Negatif - Gerçekte 1 olmayan sınıf etiketlerinin, 1 olarak tahmin edilmemesi,
- False Positive (FP): Yanlış Pozitif - Gerçekte 1 olmayan sınıf etiketlerinin, 1 olarak tahmin edilmesi,
- False Negative (FN): Yanlış Negatif - Gerçekte 1 olan sınıf etiketlerinin, 1 olarak tahmin edilmemesidir.

7.3.2. Ölçüt ifadeleri

Çalışma kapsamında, karmaşıklık matrisinden elde edilen değerlerin kullanıldığı ve model performansını değerlendirme amacıyla kullanılacak ölçüt ifadeleri ve eşitlikleri şu şekildedir:

- Model Doğruluğu (Accuracy): Öğrenme modelinin doğruluk derecesini belirler. Doğruluk değeri, Eş 7.8 ile hesaplanmaktadır.

$$Doğruluk = \frac{TP+TN}{TP+TN+FP+FN} \tag{7.8}$$

- Model Kesinliği (Precision): Öğrenme modelinin duyarlılık derecesini ölçer. Keslink değeri, Eş 7.9 ile hesaplanmaktadır.

$$Kesinlik = \frac{TP}{TP+FP} \tag{7.9}$$

- Modelin Duyarlılığı (Recall): Test sonucunda gerçek pozitif değerlerin oranı olarak bilinmektedir. Duyarlılık değeri, Eş 7.10 ile hesaplanmaktadır.

$$Duyarlılık = \frac{TP}{TP+FN} \tag{7.10}$$

- F Ölçütü (F-Score): Duyarlılık ve Kesinlik değerlerinin harmonik ortalamasıdır. F Ölçütü, Eş 7.11 ile hesaplanmaktadır.

$$F\ \text{Ölçütü} = \frac{2*Kesinlik*Duyarlılık}{Kesinlik+Duyarlılık} \tag{7.11}$$

- Hata Oranı: Hatalı ölçümlerin, toplam değer sayısına olan oranının ölçümüdür. Hata Oranı, Eş 7.12 ile hesaplanmaktadır.

$$Hata\ Oranı = \frac{FP+FN}{TP+TN+FP+FN} \tag{7.12}$$

Buraya kadar bahsedilen değerlendirme işlemleri doğrultusunda, kullanılacak olan simülasyon çalıştığı sürece elde edilecek olan veriler, ilk etapta veri tabanında sistemin çalışması süresince kayıt altına alınmış olan veriler ile seçilen algoritma kullanılarak karşılaştırılacaktır. Eğer test sonucu normal çıkarsa, yeni veriler de veri tabanına kayıt edilecek ve sistem stabil olarak çalışmaya devam edecektir. Ancak, test sonucunda yeni gelen verilerde anomali tespit edilirse, hızlı bir şekilde yöneticinin ekranında oluşan anomali hakkında uyarı verilmesi sağlanacaktır. Bu işlem sonucunda da elde edilen veriler yine veri tabanında kayıt altına alınacaktır. Anomali olsun veya olmasın her işlem sonucunda elde edilen veriler, mutlaka veri tabanına kayıt edilecektir. Çünkü, her seferinde anomali testi yapacak olan algoritma, karşılaştırma yapabilmek için bir öğrenme kümesine ihtiyaç duymaktadır. Bu çalışma mantığı ile öğrenme kümesinin sürekli olarak kendini yeni veriler ile yenilemesi ve bu sayede algoritmanın zaman geçtikçe doğruluk oranı çok daha yüksek tespitler yapması sağlanabilecektir.

Simülasyondan elde edilmiş olan veri setleri ile ilgili değerlendirme bilgileri, başlıklar halinde ayrı ayrı değerlendirilmiştir.

7.3.3. Taşıma birimi analiz bilgileri

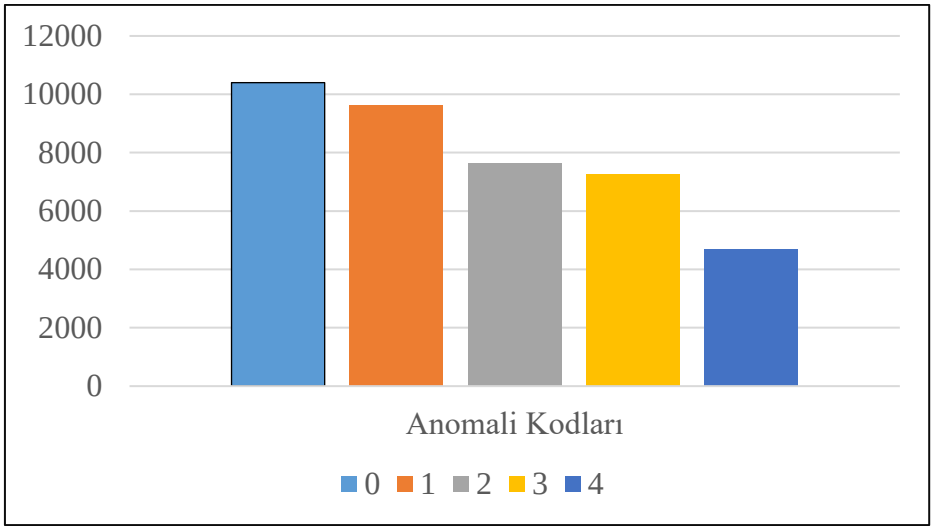
Taşıma biriminin çalıştırılması ile hem normal süreç verileri hem de anomali verileri oluşturulmuştur. Yapılan çalışmalarda, Taşıma birimi için anlık taşınan hatalı çiğ mısır miktarı ve birim içerisinde kullanılan konveyörün, anlık hareket mesafesi, şebeke gerilim değeri ve dengesiz güç harcamasına ait senaryolar kullanılmıştır. Anlık taşınan çiğ mısır

miktarı ile ilgili anomali verileri, hatasız veri setinden elde edilen standart sapma değerinin %40 oranında artırılması ile Gauss fonksiyonu kullanılarak üretilmektedir. Konveyör hareket mesafesi ile ilgili anomali verileri, anlık olarak hız hatası oluşması ile elde edilen verilerdir. Gerilim anomali verileri, şebekeden gelen gerilim verisinde oluşabilecek hatalaı verilerin, ortalama gerilim verileri içerisinde Gauss fonksiyonu kullanılarak, standart sapmanın %40 artırılması ile rastgele hesaplanmaktadır. Konveyörün dengesiz güç kullanımı sonucunda oluşan anomali verileri ise konveyörün çektiği akım değerinde aşağıda ya da yukarı yönlü rastgele olabilecek sorun sonucunda gücün dengesiz bir şekilde harcanması ile hesaplanmaktadır. Veri seti içerisinde her bir anomali değeri ayrı ayrı kodlanmıştır ve öğrenme modelleri ile oluşan anomalinin türünün de tahmin edilmesi için gerekli düzenlemeler yapılmıştır.

Taşıma Birimi, iş emrinin verilmesi ile 5000 gr'lık silolar ile çiğ mısırları depolama birimine taşıma işlemini gerçekleştirmekten sorumlu olan birimdir. Taşıma Biriminde anomali tespit süreçleri için gerekli öğrenme modelinin oluşturulması sürecinde kullanılan veri seti 39600 satırdan oluşmaktadır. Taşıma Biriminin çalışması esnasında oluşabilecek anomali türleri ve bunların veri seti içerisindeki dağılımları, Çizelge 7.4 ve Şekil 7.4'te gösterilmektedir.

Çizelge 7.4. Taşıma birimi anomali bilgileri

Anomali Kodu	**Nitelik Açıklaması**	**Anomali Verisi Dağılımları**
0	Anomali Yok	10400
1	Anlık Taşınan Ürün Miktarı Anomalisi	9620
2	Konveyör Mesafe Bilgisi Anomalisi	7632
3	Konveyör Gerilim Bilgisi Anomalisi	7254
4	Konveyör Harcanan Güç Bilgisi Anomalisi	4694



Şekil 7.4. Taşıma birimi anomali verileri dağılım grafiği

Hem hiper parametre girişi yapılmadan varsayılan ayarlarda kullanılan makine öğrenmesi algoritmaları hem de AutoML kütüphaneleri ile yapılan analizlerde, veri setinin %75'i (29700 satır) öğrenme modelinin oluşturulması için, %25'i (9900 satır) de oluşturulan öğrenme modelinin test edilerek, modelin başarım oranının hesaplanması için kullanılmıştır.

Aşağıda verilen çizelgelerde (Çizelge 7.5 – 7.14), manuel olarak standart parametreler ile kullanılan makine öğrenmesi algoritmaları ile AutoML kütüphaneleri ile elde edilen karmaşıklık matrisleri verilmiştir.

Çizelge 7.5. Taşıma birimi auto-keras kütüphanesi karmaşıklık matrisi

	Tahminler				
	0	**1**	**2**	**3**	**4**
0	1851	512	0	0	1
1	431	2097	2	0	35
2	87	79	1438	100	264
3	44	22	5	1730	0
4	109	173	174	0	746

Çizelge 7.6. Taşıma birimi auto-sklearn kütüphanesi karmaşıklık matrisi

	Tahminler				
	0	**1**	**2**	**3**	**4**
0	2379	0	3	0	23
1	0	2649	0	0	0
2	104	0	1429	85	303
3	63	0	0	1705	0
4	124	0	157	0	876

Çizelge 7.7. Taşıma birimi pycaret kütüphanesi karmaşıklık matrisi

	Tahminler				
	0	**1**	**2**	**3**	**4**
0	2395	0	2	1	23
1	0	2586	0	0	0
2	103	0	1374	103	316
3	54	0	6	1741	0
4	136	0	125	0	935

Çizelge 7.8. Taşıma birimi yapay sinir ağları algoritması karmaşıklık matrisi

	Tahminler				
	0	**1**	**2**	**3**	**4**
0	2078	55	20	177	34
1	244	1781	9	524	7
2	404	28	1097	270	169
3	212	35	62	1474	18
4	565	27	276	52	282

Çizelge 7.9. Taşıma birimi rastgele orman algoritması karmaşıklık matrisi

	Tahminler				
	0	**1**	**2**	**3**	**4**
0	2330	0	4	11	19
1	0	2565	0	0	0
2	95	0	1445	100	328
3	65	0	3	1732	1
4	124	0	197	0	881

Çizelge 7.10. Taşıma birimi karar ağacı algoritması karmaşıklık matrisi

	Tahminler				
	0	**1**	**2**	**3**	**4**
0	2115	0	95	47	107
1	0	2565	0	0	0
2	83	0	1441	104	340
3	63	0	62	1675	1
4	111	0	340	2	749

Çizelge 7.11. Taşıma birimi k_NN algoritması karmaşıklık matrisi

	Tahminler				
	0	**1**	**2**	**3**	**4**
0	1894	5	114	222	129
1	46	2492	4	12	11
2	357	2	1192	156	261
3	622	3	140	923	113
4	387	10	347	171	287

Çizelge 7.12. Taşıma birimi naive bayes algoritması karmaşıklık matrisi

	Tahminler				
	0	**1**	**2**	**3**	**4**
0	929	610	180	2	643
1	194	1308	252	0	811
2	0	631	910	53	374
3	38	13	1164	576	10
4	0	527	216	0	459

Çizelge 7.13. Taşıma birimi lojistik regresyon algoritması karmaşıklık matrisi

	Tahminler				
	0	**1**	**2**	**3**	**4**
0	1285	616	380	83	0
1	352	1771	305	1337	0
2	835	386	562	185	0
3	920	9	599	273	0
4	401	411	338	52	0

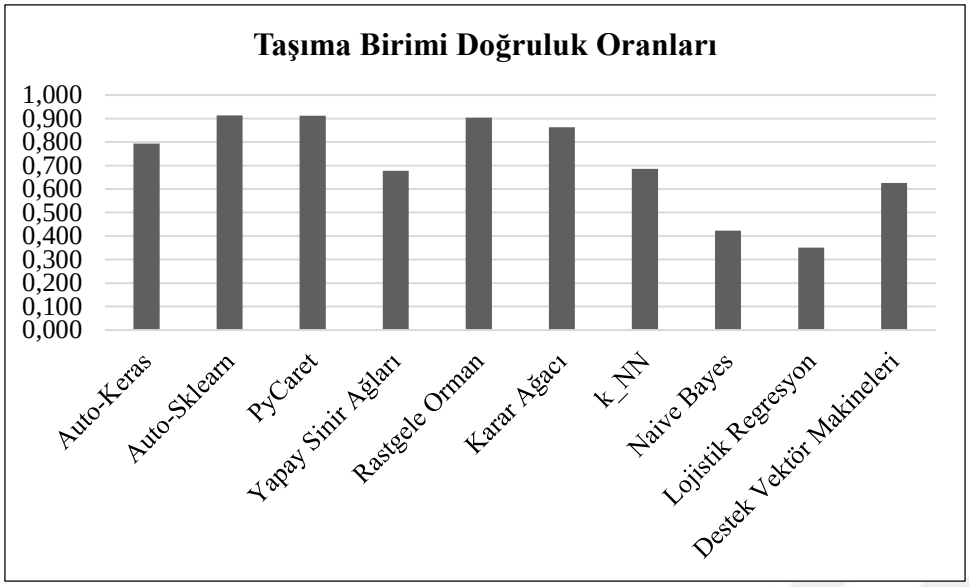
Çizelge 7.14. Taşıma birimi destek vektör makineleri algoritması karmaşıklık matrisi

	Tahminler				
	0	**1**	**2**	**3**	**4**
0	1329	744	291	0	0
1	692	4672	201	0	0
2	271	558	321	563	255
3	44	24	86	1647	0
4	233	518	351	0	100

Taşıma birimine ait veri seti için oluşturulmuş olan öğrenme modellerinin başarım oranlarının karşılaştırılması ile ilgili veriler, Çizelge 7.15 ve Şekil 7.5'te gösterilmektedir.

Çizelge 7.15. Taşıma birimi veri seti analiz sonuçları karşılaştırma çizelgesi

		Hatasız (0)	Taşınan Ürün Anomalisi (1)	Konv. Mesafe Anomali (2)	Konv. Gerilim Anomali (3)	Konv. Güç Anomali (4)	Doğruluk Oranı
Auto-Keras	**Kesinlik**	0,734	0,727	0,888	0,945	0,713	0,794
	Duyarlılık	0,783	0,818	0,731	0,961	0,621	
	F-Skoru	0,758	0,770	0,802	0,953	0,664	
Auto-Sklearn	**Kesinlik**	0,891	1,000	0,899	0,953	0,729	0,913
	Duyarlılık	0,989	1,000	0,744	0,964	0,757	
	F-Skoru	0,938	1,000	0,814	0,958	0,743	
PyCaret	**Kesinlik**	0,891	1,000	0,912	0,944	0,734	0,912
	Duyarlılık	0,989	1,000	0,725	0,967	0,782	
	F-Skoru	0,938	1,000	0,808	0,955	0,757	
Yapay Sinir Ağları	**Kesinlik**	0,593	0,925	0,749	0,590	0,553	0,678
	Duyarlılık	0,879	0,694	0,557	0,818	0,235	
	F-Skoru	0,708	0,793	0,639	0,686	0,329	
Rastgele Orman	**Kesinlik**	0,891	1,000	0,876	0,940	0,717	0,904
	Duyarlılık	0,986	1,000	0,734	0,962	0,733	
	F-Skoru	0,936	1,000	0,799	0,951	0,725	
Karar Ağacı	**Kesinlik**	0,892	1,000	0,744	0,916	0,626	0,863
	Duyarlılık	0,895	1,000	0,732	0,930	0,623	
	F-Skoru	0,893	1,000	0,738	0,923	0,624	
k_NN	**Kesinlik**	0,573	0,992	0,663	0,622	0,358	0,686
	Duyarlılık	0,801	0,972	0,606	0,512	0,239	
	F-Skoru	0,668	0,982	0,633	0,562	0,287	
Naive Bayes	**Kesinlik**	0,800	0,423	0,334	0,913	0,200	0,422
	Duyarlılık	0,393	0,510	0,462	0,320	0,382	
	F-Skoru	0,527	0,463	0,388	0,474	0,262	
Lojistik Regresyon	**Kesinlik**	0,339	0,555	0,257	0,141	0,000	0,351
	Duyarlılık	0,544	0,470	0,286	0,152	0,000	
	F-Skoru	0,417	0,509	0,271	0,146	0,000	
Destek Vektör Makineleri	**Kesinlik**	0,517	0,717	0,257	0,745	0,282	0,626
	Duyarlılık	0,562	0,840	0,163	0,914	0,083	
	F-Skoru	0,539	0,773	0,200	0,821	0,128	



Şekil 7.5. Taşıma birimi veri seti doğruluk oranları grafiği

Çizelge 7.15 ve Şekil 7.5'te gösterilen verilere göre, AutoML kütüphaneleri olan Auto-Sklearn ve PyCaret tarafından oluşturulan öğrenme modeller, diğer algoritma ve kütüphanelere göre daha başarılı sonuçlar vermişlerdir. Auto-Sklearn kütüphanesi bünyesinde en iyi sonucu veren algoritma, Şekil 7.6'da gösterilen hiper parametreler ile birlikte Rastgele Orman (RO) algoritması olmuştur. PyCaret kütüphanesinde ise en iyi sonucu veren algoritma, Şekil 7.7'de gösterilen hiper parametreler ile birlikte Light Gradient Boosting Machine (LGBM) algoritması olmuştur.

```
Rastgele Orman Sınıflama(bootstrap=False,
                         criterion="gini",
                         max_depth=None,
                         max_features=0.23832696118792362,
                         max_leaf_nodes=None,
                         min_impurity_decrease=0.0,
                         min_samples_leaf=1,
                         min_samples_split=16,
                         min_weight_fraction_leaf=0.0)
```

Şekil 7.6. Taşıma birimi veri setinde, Auto-Sklearn kütüphanesi tarafından rastgele orman algoritması için verilen hiper parametre değerleri

```
LGBM Sınıflama(boosting_type="gbdt",
               class_weight=None,
               colsample_bytree=1.0,
               importance_type="split",
               learning_rate=0.1,
               max_depth=-1,
               min_child_samples=20,
               min_child_weight=0.001,
               min_split_gain=0.0,
               n_estimators=100,
               n_jobs=-1,
               num_leaves=31,
               objective=None,
               random_state=4570,
               reg_alpha=0.0,
               reg_lambda=0.0,
               silent=True,
               subsample=1.0,
               subsample_for_bin=200000,
               subsample_freq=0)
```

Şekil 7.7. Taşıma birimi veri setinde, LGBM algoritması için pycaret kütüphanesi tarafından belirlenmiş olan hiper parametreler

Öğrenme modellerinin kayıt işlemi gerçekleştirildikten sonra, Taşıma Birimi için statik veri seti üzerinde en iyi sonucu veren RO ve LGBM algoritmaları için dinamik veri üzerinde anlık olarak anomali tespit konusundaki testler gerçekleştirilmiştir. Her test için 11000 satır veri kullanılmıştır. Sonuç olarak yapılan tüm testlere ait sonuçların ortalaması alınarak bir karara varılabilmektedir. Dinamik veri üzerinde anlık anomali tespiti için yapılan testlere ait ortalamalar, Çizelge 7.16'da verilmiştir.

Çizelge 7.16. Taşıma birimi, akan veri üzerinde yapılan testlerin ortalamaları

		Hatasız (0)	**Taşınan Ürün Anomalisi (1)**	**Konv. Mesafe Anomali (2)**	**Konv. Gerilim Anomali (3)**	**Konv. Güç Anomali (4)**	**Doğruluk Oranı**
PyCaret (LGBM)	**Kesinlik**	0,975	1,000	0,711	0,970	0,683	0,947
	Duyarlılık	0,980	0,950	0,703	0,949	0,777	
	F-Skoru	0,977	0,974	0,707	0,959	0,727	
Auto-Sklearn (RO)	**Kesinlik**	0,975	1,000	0,653	0,977	0,695	0,945
	Duyarlılık	0,978	0,950	0,720	0,956	0,751	
	F-Skoru	0,977	0,974	0,685	0,967	0,722	

Çizelge 7.16'dan da görüleceği üzere, dinamik veri üzerinde yapılan testlerin sonuçları, seçilmiş olan öğrenme modellerinin doğruluğunu ve geçerliliğini ortaya koymaktadır.

Sonuçlara bakıldığında, iki algoritma arasında ciddi bir fark olmadığı görülmektedir. Çizelgedeki sonuçlara göre, LGBM algoritması çok az bir farkla daha iyi bir sonuç verdiği görünse de algoritma tarafından oluşturulan öğrenme modeli, sistem simülasyonuna entegre edilerek, konteynır içerisinde üretim sistemi ile birlilte çalıştığında %5-10 değerleri arasında doğruluk oranı göstererek başarısız olmuştur. Bu aşamadaki olumsuz durumun sebebi olarak da LGBM öğrenme modeli konteynır içerisinde sistemin bir parçası şeklinde çalışmaya başladığında hatasız verileri tespit edemediğinden dolayı olduğu görülmüştür. Bu sebeple, Taşıma Birimine ait Anomali Tespit Cihazında Auto-Sklearn kütüphanesinden elde edilen RO algoritmasının kullanılmasına karar verilmiştir.

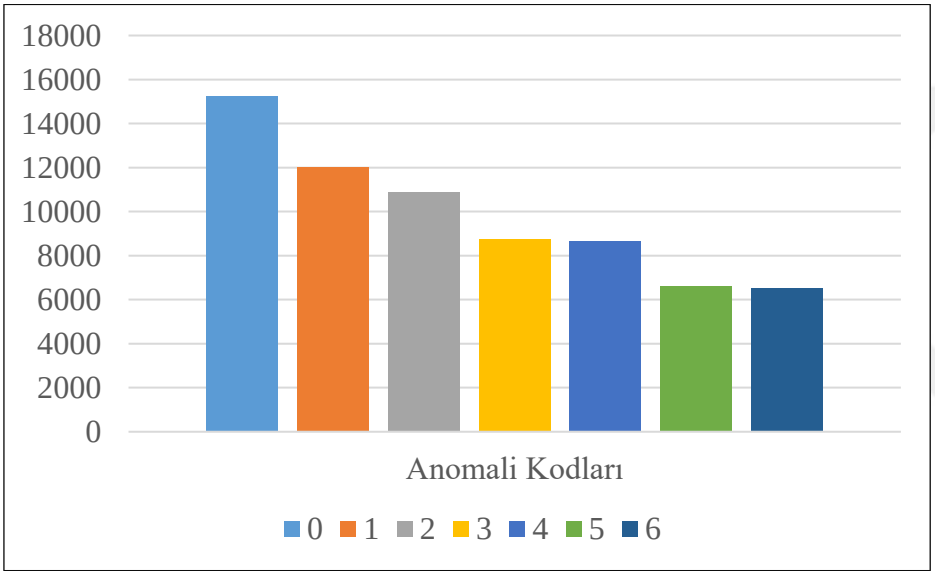
7.3.4. Dozlama birimi analiz bilgileri

Dozlama birimi veri setinde, hatasız veriler ile Çizelge 7.17'de belirtilen anomalilerin oluşmasına sebep olan veriler bulunmaktadır. Dozlama birimi ile ilgili çalışmalarda anomali verilerini oluşturmak için, huni tarafından dozlanan ürün miktarı, huni cihazının bağlı olduğu şebeke gerilim değeri, huni tarafından kullanılan güç değeri, dozlanan ürünü taşımakla görevli olan konveyörün hareket mesafesi değer, konveyörün bağlı olduğu şebeke gerilim değeri ve konveyör tarafından harcanan güç değerlerine ait senaryolar kullanılmıştır.

Dozlama Birimi, Taşıma Birimi tarafından ilk çiğ mısır partisinin depoya aktarılması ile birlikte çalışmaya başlamaktadır. Görevi, iş emrinde belirtilmiş olan üretimi yapılacak paket boyutuna uygun miktarlarda çiğ mısır dozaj işlemini gerçekleştirmektir. Dozlama Biriminde anomali tespit süreçleri için gerekli öğrenme modelinin oluşturulması sürecinde kullanılan veri seti 68716 satırdan oluşmaktadır. Dozlama Biriminin çalışması esnasında oluşabilecek anomali türleri ve bunların veri seti içerisindeki dağılımları, Çizelge 7.17 ve Şekil 7.8'de gösterilmektedir.

Çizelge 7.17. Dozlama birimi anomali bilgileri

Anomali Kodu	Nitelik Açıklaması	Anomali Verisi Dağılımları
0	Anomali Yok	15256
1	Dozlama Miktarı Anomalisi	12000
2	Huni Gerilim Bilgisi Anomalisi	10882
3	Huni Harcanan Güç Bilgisi Anomalisi	8750
4	Konveyör Mesafe Bilgisi Anomalisi	8673
5	Konveyör Gerilim Bilgisi Anomalisi	6625
6	Konveyör Harcanan Güç Bilgisi Anomalisi	6530



Şekil 7.8. Dozlama birimi anomali verileri dağılım grafiği

Dozlama birimi için hem hiper parametre girişi yapılmadan varsayılan ayarlarda kullanılan makine öğrenmesi algoritmaları hem de AutoML kütüphaneleri ile yapılan analizlerde, veri setinin %75'i (51537 satır) öğrenme modelinin oluşturulması için, %25'i (17179 satır) de oluşturulan öğrenme modelinin test edilerek, modelin başarım oranının hesaplanması için kullanılmıştır. Aşağıda verilen çizelgelerde (Çizelge 7.18 – 7.27), Dozlama birimi veri seti için varsayılan ayarlarda kullanılan makine öğrenmesi algoritmaları ve AutoML kütüphaneleri ile elde edilen karmaşıklık matrisleri bulunmaktadır.

Çizelge 7.18. Dozlama birimi auto-keras kütüphanesi karmaşıklık matrisi

	Tahminler						
	0	**1**	**2**	**3**	**4**	**5**	**6**
0	3382	0	10	2252	16	36	102
1	0	2974	0	0	0	0	0
2	87	0	2563	65	0	0	0
3	192	0	1	1958	0	0	0
4	215	0	0	0	915	20	531
5	57	0	0	0	1	2099	16
6	0	0	0	0	308	31	970

Çizelge 7.19. Dozlama birimi auto-sklearn kütüphanesi karmaşıklık matrisi

	Tahminler						
	0	**1**	**2**	**3**	**4**	**5**	**6**
0	3673	0	6	56	6	2	65
1	0	2949	0	0	0	0	0
2	12	0	2708	3	0	0	0
3	101	0	0	2112	0	0	0
4	68	0	0	0	976	0	594
5	28	0	0	0	0	2163	9
6	130	0	0	0	207	0	1311

Çizelge 7.20. Dozlama birimi pycaret kütüphanesi karmaşıklık matrisi

	Tahminler						
	0	**1**	**2**	**3**	**4**	**5**	**6**
0	3052	0	330	14	3	334	18
1	0	3039	0	0	0	0	0
2	608	0	1967	94	0	0	0
3	51	0	308	1822	0	0	0
4	76	0	0	0	970	130	480
5	454	0	0	0	0	1724	46
6	129	0	0	0	239	308	983

Çizelge 7.21. Dozlama birimi yapay sinir ağları algoritması karmaşıklık matrisi

	Tahminler						
	0	**1**	**2**	**3**	**4**	**5**	**6**
0	2391	31	985	11	5	197	178
1	94	2786	62	27	0	5	0
2	166	0	2492	57	0	0	0
3	108	18	1066	957	1	0	1
4	202	0	0	0	914	53	512
5	64	0	0	0	0	2108	1
6	382	0	0	0	202	85	1018

Çizelge 7.22. Dozlama birimi rastgele orman algoritması karmaşıklık matrisi

	Tahminler						
	0	**1**	**2**	**3**	**4**	**5**	**6**
0	3704	0	10	29	17	5	33
1	0	2974	0	0	0	0	0
2	49	0	2656	10	0	0	0
3	79	0	0	2072	0	0	0
4	66	0	0	0	1100	0	515
5	22	0	0	0	5	2135	11
6	155	0	0	0	412	0	1120

Çizelge 7.23. Dozlama birimi karar ağacı algoritması karmaşıklık matrisi

	Tahminler						
	0	**1**	**2**	**3**	**4**	**5**	**6**
0	3449	0	29	83	74	17	146
1	0	2974	0	0	0	0	0
2	41	0	2662	11	0	1	0
3	71	0	9	2070	0	1	0
4	51	0	0	0	1088	0	542
5	21	0	0	0	7	2139	6
6	133	0	0	0	563	8	983

Çizelge 7.24. Dozlama birimi k_NN algoritması karmaşıklık matrisi

	Tahminler						
	0	**1**	**2**	**3**	**4**	**5**	**6**
0	3159	34	71	262	73	61	138
1	242	2649	14	22	12	8	27
2	222	8	2385	100	0	0	0
3	435	19	97	1600	0	0	0
4	204	25	0	0	983	50	419
5	229	21	0	0	77	1761	85
6	363	44	0	0	444	135	701

Çizelge 7.25. Dozlama birimi naive bayes algoritması karmaşıklık matrisi

	Tahminler						
	0	**1**	**2**	**3**	**4**	**5**	**6**
0	0	0	87	1990	677	68	976
1	0	0	31	1500	461	16	966
2	0	0	2613	102	0	0	0
3	0	0	14	2137	0	0	0
4	0	0	0	0	951	6	724
5	0	0	0	0	12	2089	72
6	0	0	0	0	265	1	1421

Çizelge 7.26. Dozlama birimi lojistik regresyon algoritması karmaşıklık matrisi

	Tahminler						
	0	**1**	**2**	**3**	**4**	**5**	**6**
0	1462	65	998	315	170	697	91
1	806	666	746	92	21	643	0
2	501	3	1623	588	0	0	0
3	258	12	999	882	0	0	0
4	271	14	0	0	75	580	741
5	451	53	0	0	264	1290	115
6	160	22	0	0	199	515	791

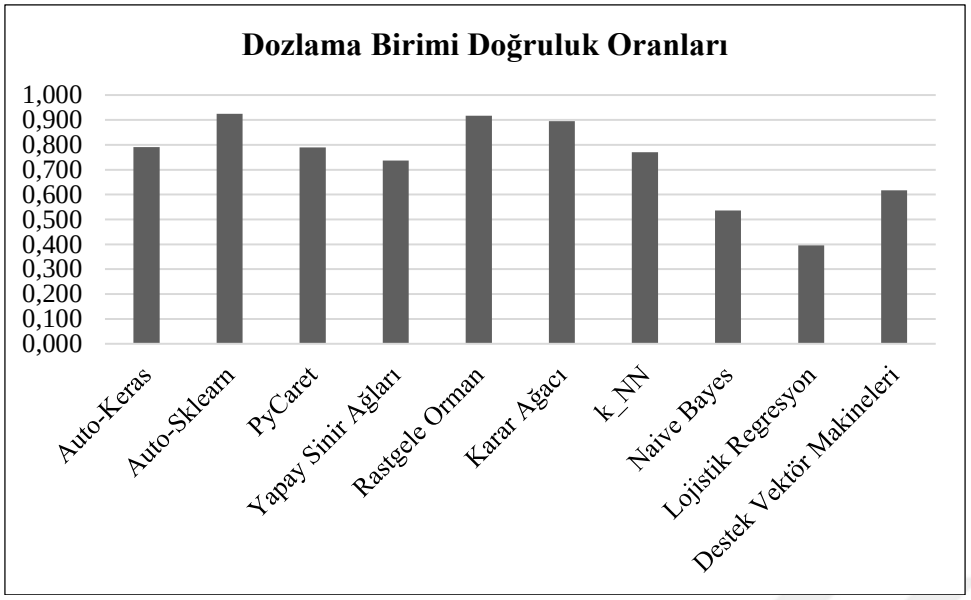
Çizelge 7.27. Dozlama birimi destek vektör makineleri algoritması karmaşıklık matrisi

	Tahminler						
	0	**1**	**2**	**3**	**4**	**5**	**6**
0	1669	867	18	916	229	18	81
1	757	1393	28	592	125	32	47
2	19	13	2638	45	0	0	0
3	232	204	13	1702	0	0	0
4	376	338	0	0	183	26	758
5	24	12	0	0	13	2117	7
6	119	380	0	0	263	20	905

Dozlama birimine ait veri seti için oluşturulmuş olan öğrenme modellerinin başarım oranlarının karşılaştırılması ile ilgili veriler, Çizelge 7.28 ve Şekil 7.9'da gösterilmektedir.

Çizelge 7.28. Dozlama birimi veri seti analiz sonuçları karşılaştırma çizelgesi

		Hatasız (0)	Dozlama Hatası (1)	Huni Gerilim Hatası (2)	Huni Güç Hatası (3)	Konv. Mesafe Hatası (4)	Konv. Gerilim Hatası (5)	Konv. Güç Hatası (6)	Doğruluk Oranı
Auto-Keras	**Kesinlik**	0,860	1,000	0,996	0,458	0,738	0,960	0,599	0,790
	Duyarlılık	0,583	1,000	0,944	0,910	0,544	0,966	0,741	
	F-Skoru	0,695	1,000	0,969	0,609	0,626	0,963	0,663	
Auto-Sklearn	**Kesinlik**	0,916	1,000	0,998	0,973	0,821	0,999	0,662	0,925
	Duyarlılık	0,965	1,000	0,994	0,954	0,596	0,983	0,796	
	F-Skoru	0,939	1,000	0,996	0,964	0,690	0,991	0,723	
PyCaret	**Kesinlik**	0,698	1,000	0,755	0,944	0,800	0,691	0,644	0,789
	Duyarlılık	0,814	1,000	0,737	0,835	0,586	0,775	0,593	
	F-Skoru	0,752	1,000	0,746	0,886	0,676	0,731	0,617	
Yapay Sinir Ağları	**Kesinlik**	0,702	0,983	0,541	0,910	0,815	0,861	0,595	0,737
	Duyarlılık	0,630	0,937	0,918	0,445	0,544	0,970	0,603	
	F-Skoru	0,664	0,959	0,681	0,598	0,652	0,912	0,599	
Rastgele Orman	**Kesinlik**	0,909	1,000	0,996	0,982	0,717	0,998	0,667	0,917
	Duyarlılık	0,975	1,000	0,978	0,963	0,654	0,983	0,664	
	F-Skoru	0,941	1,000	0,987	0,972	0,684	0,990	0,665	
Karar Ağacı	**Kesinlik**	0,916	1,000	0,986	0,957	0,628	0,988	0,586	0,894
	Duyarlılık	0,908	1,000	0,980	0,962	0,647	0,984	0,583	
	F-Skoru	0,912	1,000	0,983	0,959	0,638	0,986	0,584	
k_NN	**Kesinlik**	0,651	0,946	0,929	0,806	0,619	0,874	0,512	0,771
	Duyarlılık	0,832	0,891	0,878	0,744	0,585	0,810	0,416	
	F-Skoru	0,730	0,918	0,903	0,774	0,601	0,841	0,459	
Naive Bayes	**Kesinlik**	0,000	0,000	0,952	0,373	0,402	0,958	0,342	0,536
	Duyarlılık	0,000	0,000	0,962	0,993	0,566	0,961	0,842	
	F-Skoru	0,000	0,000	0,957	0,542	0,470	0,960	0,486	
Lojistik Regresyon	**Kesinlik**	0,374	0,798	0,372	0,470	0,103	0,346	0,455	0,395
	Duyarlılık	0,385	0,224	0,598	0,410	0,045	0,594	0,469	
	F-Skoru	0,379	0,350	0,458	0,438	0,062	0,437	0,462	
Destek Vektör Makineleri	**Kesinlik**	0,522	0,434	0,978	0,523	0,225	0,957	0,503	0,617
	Duyarlılık	0,439	0,468	0,972	0,791	0,109	0,974	0,536	
	F-Skoru	0,477	0,451	0,975	0,630	0,147	0,965	0,519	



Şekil 7.9. Dozlama birimi veri seti doğruluk oranları grafiği

Çizelge 7.28 ve Şekil 7.9'da gösterilen verilere göre, Dozlama birimi için en iyi iki sonuç Auto-Sklearn kütüphanesi ve Rastgele Orman algoritması tarafından alınmıştır. Auto-Sklearn kütüphanesi bünyesinde en iyi sonucu veren algoritma, Şekil 7.10'da gösterilen hiper parametreler ile birlikte Rastgele Orman (RO) algoritması olmuştur. Veri seti üzerinde en iyi ikinci sonucu veren, varsayılan ayarlar ile kullanılan RO algoritmasına ait kod dizisi ise Şekil 7.11'de verilmiştir.

```
Rastgele Orman Sınıflama(bootstrap=False,
                         criterion="gini",
                         max_depth=None,
                         max_features=0.23832696118792362,
                         max_leaf_nodes=None,
                         min_impurity_decrease=0.0,
                         min_samples_leaf=1,
                         min_samples_split=16,
                         min_weight_fraction_leaf=0.0)
```

Şekil 7.10. Dozlama birimi veri setinde, auto-sklearn kütüphanesi tarafından rastgele orman algoritması için verilen hiper parametre değerleri

```
Rastgele Orman Sınıflama(n_estimators=100, random_state=42)
```

Şekil 7.11. Dozlama birimi veri setinde, varsayılan ayarlarda kullanılan rastgele orman algoritması modellemesi

Öğrenme modellerinin kayıt işlemi gerçekleştirildikten sonra, Dozlama Birimi için statik veri seti üzerinde en iyi sonucu veren Auto-Sklearn (RO) ve RO algoritmaları için dinamik

veri üzerinde anlık olarak anomali tespit konusundaki testler gerçekleştirilmiştir. Her test için 10000 satır veri kullanılmıştır. Sonuç olarak yapılan tüm testlere ait sonuçların ortalaması alınarak bir karara varılabilmektedir. Dinamik veri üzerinde anlık anomali tespiti için yapılan testlere ait ortalamalar, Çizelge 7.29'da verilmiştir.

Çizelge 7.29 Dozlama birimi, akan veri üzerinde yapılan testlerin ortalamaları

		Hatasız (0)	Dozlama Hatası (1)	Huni Gerilim Hatası (2)	Huni Güç Hatası (3)	Konv. Mesafe Hatası (4)	Konv. Gerilim Hatası (5)	Konv. Güç Hatası (6)	Doğruluk Oranı
Auto-Sklearn (RO)	**Kesinlik**	0,934	0,800	0,997	0,870	0,749	0,996	0,648	0,917
	Duyarlılık	0,991	0,610	0,976	0,947	0,618	0,976	0,682	
	F-Skoru	0,962	0,692	0,987	0,907	0,677	0,986	0,664	
RO	**Kesinlik**	0,934	0,800	0,998	0,870	0,697	0,996	0,603	0,912
	Duyarlılık	0,984	0,616	0,976	0,943	0,629	0,976	0,638	
	F-Skoru	0,958	0,696	0,987	0,905	0,662	0,985	0,620	

Çizelge 7.29'dan da görüleceği üzere, dinamik veri üzerinde yapılan testlerin sonuçları, seçilmiş olan öğrenme modellerinin doğruluğunu ve geçerliliğini ortaya koymaktadır. Sonuçlara bakıldığında, iki algoritma arasında ciddi bir fark olmadığı görülmektedir. Buradaki sonuçlara göre Auto-Sklearn (RO) algoritması çok az bir farkla daha iyi sonuç verdiği görülmektedir. Bu sebeple, Dozlama Birimine ait Anomali Tespit Cihazında Auto-Sklearn kütüphanesinden elde edilen RO algoritmasının kullanılmasına karar verilmiştir.

7.3.5. Üretim birimi analiz bilgileri

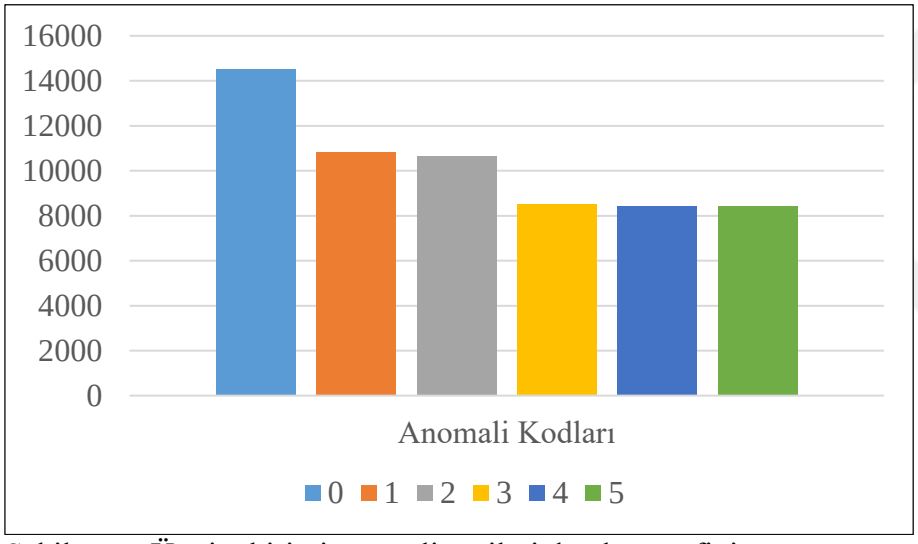
Üretim birimine veri seti, hatasız veriler ile Çizelge 7.30'da belirtilen anomalilerin oluşmasına sebep olan veriler bulunmaktadır. Üretim birimi ile ilgili çalışmalarda anomali verilerini oluşturmak için, fırının bağlı olduğu şebeke üzerindeki gerilim bilgisi, fırın tarafından anlık tüketilen güç değeri, üretilen ürünü taşımakla görevli olan konveyörün hareket mesafesi değer, konveyörün bağlı olduğu şebeke gerilim değeri ve konveyör tarafından harcanan güç değerlerine ait senaryolar kullanılmıştır.

Üretim Birimi, Dozlama Birimi tarafından ilk paket için gerekli dozaj işleminin tamamlanması ile birlikte çalışmaya başlamaktadır. Görevi, iş emrinde belirtilen paket boyutuna göre dozajı tamamlanmış olan ürünü işleyip paketleyerek üretim sürecinin tamamlanmasını sağlamaktır. Üretim Biriminde anomali tespit süreçleri için gerekli öğrenme modelinin oluşturulması sürecinde kullanılan veri seti 61336 satırdan

oluşmaktadır. Üretim Biriminin çalışması esnasında oluşabilecek anomali türleri ve bunların veri seti içerisindeki dağılımları, Çizelge 7.30 ve Şekil 7.12'de gösterilmektedir.

Çizelge 7.30. Üretim birimi anomali bilgileri

Anomali Kodu	Nitelik Açıklaması	Anomali Verisi Dağılımları
0	Anomali Yok	14492
1	Fırın Gerilim Bilgisi Anomalisi	10836
2	Fırın Harcanan Güç Bilgisi Anomalisi	10657
3	Konveyör Mesafe Bilgisi Anomalisi	8509
4	Konveyör Gerilim Bilgisi Anomalisi	8426
5	Konveyör Harcanan Güç Bilgisi Anomalisi	8416



Şekil 7.12. Üretim birimi anomali verileri dağılım grafiği

Üretim birimi için hem hiper parametre girişi yapılmadan varsayılan ayarlarda kullanılan makine öğrenmesi algoritmaları hem de AutoML kütüphaneleri ile yapılan analizlerde, veri setinin %75'i (46002 satır) öğrenme modelinin oluşturulması için, %25'i (15334 satır) de oluşturulan öğrenme modelinin test edilerek, modelin başarım oranının hesaplanması için kullanılmıştır. Aşağıda verilen çizelgelerde (Çizelge 7.31 – 7.40), Üretim birimi veri seti için varsayılan ayarlarda kullanılan makine öğrenmesi algoritmaları ve AutoML kütüphaneleri ile elde edilen karmaşıklık matrisleri bulunmaktadır.

Çizelge 7.31. Üretim birimi auto-keras kütüphanesi karmaşıklık matrisi

	Tahminler					
	0	**1**	**2**	**3**	**4**	**5**
0	3552	3	0	49	11	0
1	66	2586	0	0	0	0
2	0	0	2700	0	0	0
3	237	0	0	1906	2	0
4	85	0	0	6	1978	0
5	0	0	0	0	0	2153

Çizelge 7.32. Üretim birimi auto-sklearn kütüphanesi karmaşıklık matrisi

	Tahminler					
	0	**1**	**2**	**3**	**4**	**5**
0	3583	0	0	21	0	0
1	63	2621	0	0	0	0
2	0	0	2726	0	0	0
3	236	0	0	1878	0	0
4	37	0	0	2	2064	0
5	0	0	0	0	0	2103

Çizelge 7.33. Üretim Birimi pycaret Kütüphanesi Karmaşıklık Matrisi

	Tahminler					
	0	**1**	**2**	**3**	**4**	**5**
0	3211	0	0	13	421	0
1	7	2595	0	0	0	0
2	0	0	2749	0	0	0
3	111	0	0	1780	194	1
4	170	0	0	19	1978	0
5	0	0	0	0	0	2085

Çizelge 7.34. Üretim birimi yapay sinir ağları algoritması karmaşıklık matrisi

	Tahminler					
	0	**1**	**2**	**3**	**4**	**5**
0	2529	129	736	47	174	0
1	269	2080	303	0	0	0
2	851	28	1821	0	0	0
3	202	0	0	1903	40	0
4	61	0	0	6	2002	0
5	0	0	0	0	0	2153

Çizelge 7.35. Üretim birimi rastgele orman algoritması karmaşıklık matrisi

	Tahminler					
	0	**1**	**2**	**3**	**4**	**5**
0	3586	0	0	28	1	0
1	49	2602	1	0	0	0
2	0	0	2700	0	0	0
3	183	0	0	1962	0	0
4	47	0	0	5	2017	0
5	0	0	0	0	0	2153

Çizelge 7.36. Üretim birimi karar ağacı algoritması karmaşıklık matrisi

	Tahminler					
	0	**1**	**2**	**3**	**4**	**5**
0	3418	5	0	172	20	0
1	15	2637	0	0	0	0
2	2	0	2698	0	0	0
3	129	0	0	2010	6	0
4	27	0	0	4	2038	0
5	0	0	0	0	0	2153

Çizelge 7.37. Üretim birimi k_NN algoritması karmaşıklık matrisi

	Tahminler					
	0	**1**	**2**	**3**	**4**	**5**
0	2650	99	190	37	639	0
1	125	2517	10	0	0	0
2	406	27	2267	0	0	0
3	294	0	0	1478	373	0
4	577	0	0	45	1447	0
5	0	0	0	0	0	2153

Çizelge 7.38. Üretim birimi naive bayes algoritması karmaşıklık matrisi

	Tahminler					
	0	**1**	**2**	**3**	**4**	**5**
0	0	288	2138	1142	47	0
1	0	2607	45	0	0	0
2	0	89	2611	0	0	0
3	0	0	0	2131	14	0
4	0	0	0	61	2008	0
5	0	0	0	0	0	2153

Çizelge 7.39. Üretim birimi lojistik regresyon algoritması karmaşıklık matrisi

	Tahminler					
	0	**1**	**2**	**3**	**4**	**5**
0	237	1479	714	466	719	0
1	36	1834	782	0	0	0
2	230	673	1797	0	0	0
3	267	2	0	961	915	0
4	0	0	0	462	1607	0
5	0	0	0	0	0	2153

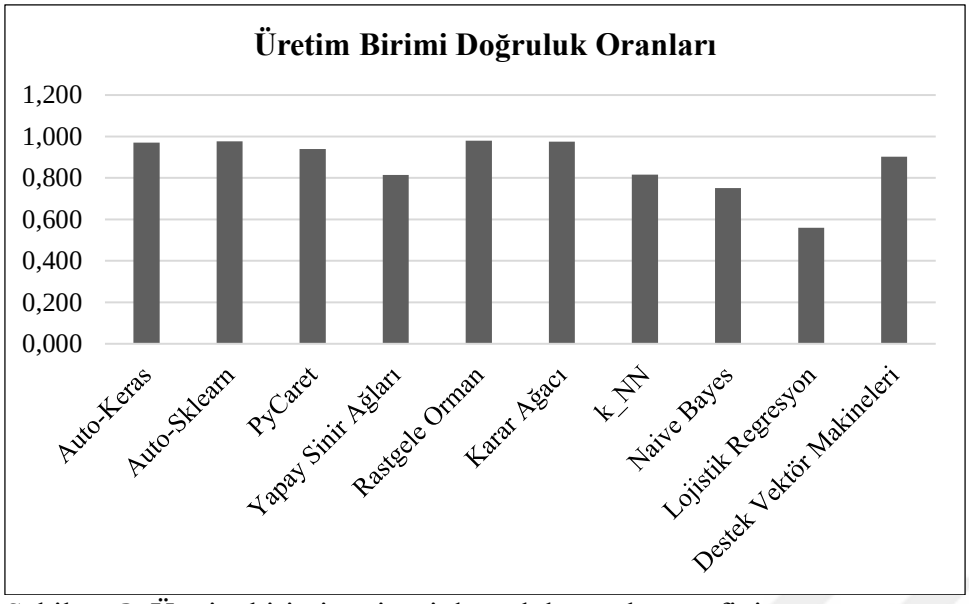
Çizelge 7.40. Üretim birimi destek vektör makineleri algoritması karmaşıklık matrisi

	Tahminler					
	0	**1**	**2**	**3**	**4**	**5**
0	2572	7	0	1036	0	0
1	65	2587	0	0	0	0
2	0	0	2700	0	0	0
3	298	0	0	1847	0	0
4	17	0	0	64	1988	0
5	0	0	0	0	0	2153

Üretim birimine ait veri seti için oluşturulmuş olan öğrenme modellerinin başarım oranlarının karşılaştırılması ile ilgili veriler, Çizelge 7.41 ve Şekil 7.13'te gösterilmektedir.

Çizelge 7.41. Üretim birimi veri seti ayrıntılı test sonuçları

		Hatasız (0)	Fırın Gerilim Hatası (1)	Fırın Güç Hatası (2)	Konv. Mesafe Hatası (4)	Konv. Gerilim Hatası (5)	Konv. Güç Hatası (6)	Doğruluk Oranı
Auto-Keras	**Kesinlik**	0,902	0,999	1,000	0,972	0,993	1,000	0,970
	Duyarlılık	0,983	0,975	1,000	0,889	0,956	1,000	
	F-Skoru	0,940	0,987	1,000	0,928	0,974	1,000	
Auto-Sklearn	**Kesinlik**	0,914	1,000	1,000	0,988	1,000	1,000	0,977
	Duyarlılık	0,994	0,977	1,000	0,888	0,981	1,000	
	F-Skoru	0,953	0,988	1,000	0,935	0,991	1,000	
PyCaret	**Kesinlik**	0,918	1,000	1,000	0,982	0,763	1,000	0,939
	Duyarlılık	0,881	0,997	1,000	0,853	0,913	1,000	
	F-Skoru	0,899	0,999	1,000	0,913	0,831	1,000	
Yapay Sinir Ağları	**Kesinlik**	0,646	0,930	0,637	0,973	0,903	1,000	0,814
	Duyarlılık	0,700	0,784	0,674	0,887	0,968	1,000	
	F-Skoru	0,672	0,851	0,655	0,928	0,934	1,000	
Rastgele Orman	**Kesinlik**	0,928	1,000	1,000	0,983	1,000	1,000	0,980
	Duyarlılık	0,992	0,981	1,000	0,915	0,975	1,000	
	F-Skoru	0,959	0,990	1,000	0,948	0,987	1,000	
Karar Ağacı	**Kesinlik**	0,952	0,998	1,000	0,919	0,987	1,000	0,975
	Duyarlılık	0,946	0,994	0,999	0,937	0,985	1,000	
	F-Skoru	0,949	0,996	1,000	0,928	0,986	1,000	
k_NN	**Kesinlik**	0,654	0,952	0,919	0,947	0,588	1,000	0,816
	Duyarlılık	0,733	0,949	0,840	0,689	0,699	1,000	
	F-Skoru	0,691	0,951	0,877	0,798	0,639	1,000	
Naive Bayes	**Kesinlik**	0,000	0,874	0,545	0,639	0,971	1,000	0,751
	Duyarlılık	0,000	0,983	0,967	0,993	0,971	1,000	
	F-Skoru	0,000	0,000	0,697	0,778	0,971	1,000	
Lojistik Regresyon	**Kesinlik**	0,308	0,460	0,546	0,509	0,496	1,000	0,560
	Duyarlılık	0,066	0,692	0,666	0,448	0,777	1,000	
	F-Skoru	0,108	0,552	0,600	0,476	0,605	1,000	
Destek Vektör Makineleri	**Kesinlik**	0,871	0,997	1,000	0,627	1,000	1,000	0,903
	Duyarlılık	0,711	0,975	1,000	0,861	0,961	1,000	
	F-Skoru	0,783	0,986	1,000	0,725	0,980	1,000	



Şekil 7.13. Üretim birimi veri seti doğruluk oranları grafiği

Çizelge 7.41 ve Şekil 7.13'te gösterilen verilere göre, Üretim birimi için en iyi iki sonuç Auto-Sklearn kütüphanesi ve Rastgele Orman algoritması tarafından alınmıştır. Auto-Sklearn kütüphanesi bünyesinde en iyi sonucu veren algoritma, Şekil 7.14'te gösterilen hiper parametreler ile birlikte Rastgele Orman (RO) algoritması olmuştur. Veri seti üzerinde en iyi ikinci sonucu veren, varsayılan ayarlar ile kullanılan RO algoritmasına ait kod dizisi ise Şekil 7.15'te verilmiştir.

```
Rastgele Orman Sınıflama(bootstrap=True,
                         criterion="gini",
                         max_depth=None,
                         max_features=0.5,
                         max_leaf_nodes=None,
                         min_impurity_decrease=0.0,
                         min_samples_leaf=1,
                         min_samples_split=2,
                         min_weight_fraction_leaf=0.0)
```

Şekil 7.14. Üretim birimi veri setinde, auto-sklearn kütüphanesi tarafından rastgele orman algoritması için verilen hiper parametre değerleri

```
Rastgele Orman Sınıflama(n_estimators=100, random_state=42)
```

Şekil 7.15. Üretim birimi veri setinde, varsayılan ayarlarda kullanılan rastgele orman algoritması modellemesi

Öğrenme modellerinin kayıt işlemi gerçekleştirildikten sonra, Üretim Birimi için statik veri seti üzerinde en iyi sonucu veren Auto-Sklearn (RO) ve RO algoritmaları için dinamik veri

üzerinde anlık olarak anomali tespit konusundaki testler gerçekleştirilmiştir. Her test için 10000 satır veri kullanılmıştır. Sonuç olarak yapılan tüm testlere ait sonuçların ortalaması alınarak bir karara varılabilmektedir. Dinamik veri üzerinde anlık anomali tespiti için yapılan testlere ait ortalamalar, Çizelge 7.42'de verilmiştir.

Çizelge 7.42. Üretim birimi, akan veri üzerinde yapılan testlerin ortalamaları

		Hatasız (0)	**Fırın Gerilim Hatası (1)**	**Fırın Güç Hatası (2)**	**Konv. Mesafe Hatası (4)**	**Konv. Gerilim Hatası (5)**	**Konv. Güç Hatası (6)**	**Doğruluk Oranı**
Auto-Sklearn (RO)	**Kesinlik**	0,954	1,000	1,000	0,772	1,000	1,000	0,951
	Duyarlılık	0,998	0,985	0,695	0,907	0,978	1,000	
	F-Skoru	0,976	0,992	0,820	0,834	0,989	1,000	
RO	**Kesinlik**	0,930	1,000	1,000	0,765	1,000	1,000	0,938
	Duyarlılık	0,991	0,905	0,704	0,905	0,978	1,000	
	F-Skoru	0,959	0,950	0,826	0,829	0,989	1,000	

Çizelge 7.42'dan da görüleceği üzere, dinamik veri üzerinde yapılan testlerin sonuçları, seçilmiş olan öğrenme modellerinin doğruluğunu ve geçerliliğini ortaya koymaktadır. Sonuçlara bakıldığında, Auto-Sklearn (RO) algoritmasının daha iyi sonuç verdiği görülmektedir. Bu sebeple, Üretim Birimine ait Anomali Tespit Cihazında Auto-Sklearn kütüphanesinden elde edilen RO algoritmasının kullanılmasına karar verilmiştir.

8. SONUÇ VE TARTIŞMA

Zeki fabrika yapılarında, gerçekleştirilebilecek çeşitli siber saldırılar ve sistem çalışması esnasında oluşabilecek bazı dengesiz çalışma süreçleri sonucunda, cihazlar arasında akan verilerde bozulma meydana gelebilir. Bu durum, sistemin hatalı çalışmasına, hatalı çıktının ortaya çıkmasına ve süreçlerin tamamen durmasına sebep olabilmektedir. Sistem kurulumlarında bu ihtimallerin mutlak suretle göz önüne alınması gerekmektedir. Bundan dolayı sistem içerisinde çalışan cihazlar arasında gerçekleştirecekleri veri iletişimi daimi olarak kontrol altında tutulması gerekmektedir. Uygulanan her süreç sonrasında elde edilen veriler, mevcut veri tabanındaki veriler ile makine öğrenmesi algoritmaları kullanılarak karşılaştırılabilir. Böylelikle süreçlerde, bir saldırı sonucu veya bir hata sonucunda oluşacak anomali, vakit kaybetmeden tespit edilerek yöneticilerin durum hakkında bilgilendirmeleri sağlanabilir.

Çalışma kapsamında, MQTT alt yapısını kullanan bir zeki fabrika birimlerine örnek bir patlamış mısır üretim sistemi simülasyonu kurulmuştur. Bu simülasyon, asıl sistemde uygulanan veri iletim süreçlerinin örneklerini oluşturmaktadır. Bu sayede simülasyondan üretilmiş ve öğrenme modellerinin oluşturabilmesi için gerekli veri setlerinin üretimi gerçekleştirilmiştir. Simülasyon içerisinde bulunan birimlerden, Taşıma birimi, Dozlama birimi ve Üretim birimine ait hatasız veriler ve anomali verilerinden oluşan veri setleri elde edilerek çalışmada belirtilmiş olan makine öğrenmesi algoritmaları ve AutoML kütüphaneleri ile analizler yapılarak, her biri için ayrı ayrı öğrenme modelleri oluşturulmuş ve sonuçlar karşılaştırılmıştır. Elde edilen bulgular doğrultusunda, öğrenme modellerinin oluşturulması kapsamında;

- Taşma birimi veri setinde Auto-Sklearn kütüphanesi tarafından belirlenen hiper parametre değerleri ile RO algoritması %91,3; PyCaret kütüphanesi tarafından belirlenen hiper parametre değerleri ile LGBM algoritması %91,2,
- Dozlama birimi veri setinde Auto-Sklearn kütüphanesi tarafından belirlenen hiper parametre değerleri ile RO algoritması %92,5; varsayılan hiper parametre değerleri ile kullanılan RO algoritması %91,7,

- Üretim birimi veri setinde Auto-Sklearn kütüphanesi tarafından belirlenen hiper parametre değerleri ile RO algoritması %97,7; varsayılan hiper parametre değerleri ile kullanılan RO algoritması %98,

doğruluk oranları ile başarımı en yüksek öğrenme modellerini oluşturmuşlardır.

Çalışmanın devam eden sürecinde her veri seti için, belirlenmiş olan öğrenme modellerinin dinamik veri üzerinde gerçek zamanlı olarak analizler yapılmıştır. Dinamik veri üzerinde, her bir öğrenme modeli için kararlı değerlerin ortaya çıkartılabilmesi için beşer defa analizler yapılarak, sonuçlar kayıt edilmiştir. Dinamik veri üzerinde yapılan gerçek zamanlı analizlerde, her seferde Taşıma birimi için 11000 satır, Dozlama birimi ve Üretim birimi için de 10000 satır veri test sonuçlarının belirlenmesi için kullanılmıştır. Yapılan analizler sonucunda, öğrenme modelleri için gerçek zamanlı test sonuçlarının ortalamaları ve simülasyonda kullanılacak model tercihleri aşağıda belirtilmiştir:

- Taşma birimi veri setinde Auto-Sklearn kütüphanesi tarafından belirlenen hiper parametre değerleri ile RO algoritması %94,5; PyCaret kütüphanesi tarafından belirlenen hiper parametre değerleri ile LGBM algoritması %94,7 oranlarında başarım göstermişlerdir. PyCaret kütüphanesi tarafından önerilen LGBM algoritması, simülasyonun tam kapasite çalıştırılması esnasında kullanımında stabil çalışmadığı tespit edildiğinden dolayı, Auto-Sklearn kütüphanesi tarafından önerilmiş olan RO algoritması ile oluşturulan öğrenme modelinin simülasyonda kullanılmasına karar verilmiştir.
- Dozlama birimi veri setinde Auto-Sklearn kütüphanesi tarafından belirlenen hiper parametre değerleri ile RO algoritması %91,7; varsayılan hiper parametre değerleri ile kullanılan RO algoritması %91,2 oranlarında başarım göstermişlerdir. İki modelin değerleri de birbirlerine çok yakın seviyede elde edilmiştir. Sonuç olarak, elde edilen değerler çerçevesinde Auto-Sklearn kütüphanesi tarafından önerilmiş olan RO algoritması ile oluşturulan öğrenme modelinin simülasyonda kullanılmasına karar verilmiştir.
- Üretim birimi veri setinde Auto-Sklearn kütüphanesi tarafından belirlenen hiper parametre değerleri ile RO algoritması %95,1; varsayılan hiper parametre değerleri ile kullanılan RO algoritması %93,8 oranlarında başarım göstermişlerdir. Sonuç

olarak, elde edilen değerler çerçevesinde Auto-Sklearn kütüphanesi tarafından önerilmiş olan RO algoritması ile oluşturulan öğrenme modelinin simülasyonda kullanılmasına karar verilmiştir.

Analizlerin tamamlanması ve simülasyonun çalıştırılması esnasında anomali tespit için kullanılacak olan öğrenme modellerinin belirlenmesinden sonra, çalışma için üretilmiş olan simülasyona, anomali tespit sürecini yönetecek modüllerin eklenmesi sağlanmıştır. Bu modüller, simülasyon çalıştığı süre boyunca ortaya çıkan verileri, gerçek zamanlı olarak satır satır inceleyerek, veriler üzerinde herhangi bir anomali olup olmadığını tespit etmeye çalışmakta ve anomali tespit edildiğinde, hangi tür anomalinin ortaya çıktığını yakalamaya çalışmaktadır. Sistem oluşan anomalilerin, denetçi tarafından takip edilebilmesi ve gerekli müdahalelerin yapılabilmesi için de simülasyona bir anomali takip ekranı eklenmiştir. Anomali tespiti yapılan her bir makine için, sistem çalıştırılmaya başlandığında, anomali tespit cihazı, son 10 saniye içerisinde aynı anomali tipinden en az 20 adet tespit ettiği takdirde bunu, anomali takip ekranına gönderilerek, denetçiye bildirilmektedir. Bu sınırın altında kalan tespitler için ise herhangi bir bildirim yapılmamaktadır.

Ayrıca sistem, kullanıcı tarafından yapılan veri girişlerinin takibini de gerçekleştirebilmektedir. Bu sayede, eğer simülasyon dışarıdan yapılabilecek kontrolsüz bir müdahale sonucunda, kullanıcının yapmış olduğu veri girişine aykırı bir şekilde çalışmayı sürdürürse, sistem buna müdahale ederek anomali takip ekranı vasıtası ile denetçiye gerekli bildirimleri de yapabilecektir.

Sonuç olarak, Endüstri 4.0 kapsamında bir anomali tespit sistemi oluşturulabilmesi için, otonom üretim sistemlerine örnek olması açısından bir patlamış mısır üretim sistemi simülasyonu geliştirilmiştir. Bu tür bir anomali tespit sisteminin oluşturulması amacıyla gerçek verilerin elde edilmesindeki zorluklardan dolayı, sentetik veriler üzerinde çalışmaya karar verildiği için simülasyon yöntemi kullanılmıştır. Oluşturulan simülasyon, cihazların çalışma esnasında oluşturdukları verileri kayıt ederek, gerçek zamanlı olarak anomali analizine tabi tutabilmektedir. Ayrıca tespit ettiği anomalinin türünü de belirleyerek, belirlenmiş olan limitin üstüne çıkan anomali türünü, cihaz bilgisi ve anomali türü bilgisi ile beraber takip ekranı vasıtası ile denetçiye bildirebilmektedir.

Geliştirilmiş olan bu sistem ile otonom üretim sistemlerinde oluşabilecek anomalilerin yine otonom bir yapıda hızlı bir şekilde tespit edilebileceği gösterilmiştir. Ayrıca anomali tespit sistemi uygulamalarında çok sınıflı öğrenme modellerinin kullanılması sayesinde anomalilerin çeşitlendirilmesi ile birlikte anomalinin türünün ve gerçekleştiği cihaz bilgisinin de tespit edilebileceği gösterilmiştir.

Bu çalışmada sentetik verilerin kullanımının tercih edilmesi sebebiyle, ileriki çalışmalar için gerçek veri setleri üzerinde çok sınıflı bir anomali tespit sisteminin geliştirilmesi önerilebilmektedir. Ayrıca hem farklı türlerde anomalileri tespit edecek, hem de anomaliyi oluşmadan önce tahmin ederek öncesinde önlem alınmasını sağlayabilecek bir uygulamanın geliştirilmesine yönelik çalışamaların yapılması da önerilebilmektedir.

KAYNAKLAR

1. Frank A. G., Dalenogare L. S., and Ayala N. F. (2019), Industry 4.0 technologies: Implementation patterns in manufacturing companies, *International Journal of Production Economics*, 210, 15–26.

2. Kumar R., Gupta P., Singh S., and Jain D. (2021), Implementation of Industry 4.0 Practices in Indian Organization: A Case Study, In R. K. Phanden, K. Mathiyazhagan, R. Kumar, and J. P. Davim, (Eds.) , *Advances in Industrial and Production Engineering*, Singapore: Springer, pp. 35–44.

3. Sayilgan E. and Isler Y. (2017, Ekim), *Medical devices sector in medical industry 4.0*, Paper presented at the 2017 Medical Technologies National Conference, Trabzon.

4. Weyer S., Schmitt M., Ohmer M., and Gorecky D. (2015), Towards Industry 4.0 - Standardization as the crucial challenge for highly modular, multi-vendor production systems, *IFAC-PapersOnLine*, 48(3), 579–584.

5. Mehrotra K. G., Mohan C. K., and Huang H. (2017), *Anomaly Detection Algorithms and Principles.* Cham, Switzerland: Springer International Publishing.

6. Ahmed M., Mahmood A. N., and Islam M. R. (2016), A survey of anomaly detection techniques in financial domain, *Future Generation Computer Systems*, 55, 278–288.

7. Agrawal S. and Agrawal J. (2015), Survey on Anomaly Detection using Data Mining Techniques, *Procedia Computer Science*, 60(1), 708–713.

8. Gürsakal N. (2018), *Makine Öğrenmesi*. Bursa: Dora Yayıncılık, 63, 230-232.

9. Kang M. and Jameson N. J. (2018), Machine Learning: Fundamentals, In M. G. Pecht and M. Kang, (Eds.), *Prognostics and Health Management of Electronics: Fundamentals, Machine Learning, and the Internet of Things*, Chichester, UK: Wiley-IEEE Press, pp. 85–109.

10. Sayer S. and Ülker A. (2014), Ürün Yaşam Döngüsü Yönetimi, *Mühendis ve Makina*, 55(657), 65–72.

11. Can A. V. ve Kıymaz M. (2016), Bilişim Teknolojilerinin Perakende Mağazacılık Sektörüne Yansımaları: Muhasebe Departmanlarında Endüstri 4.0 Etkisi, *Süleyman Demirel Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, CİEP Özel, 107–117.

12. Coşkun S., Gençay E., and Kayıkcı Y. (2016, Ekim), *Adapting Engineering Education to Industrie*, Paper presented at the 16th Production Research Symposium, İstanbul

13. Akben İ. ve Avşar İ. İ. (2018), Endüstri 4.0 ve Karanlık Üretim: Genel Bir Bakış, *Türk Sosyal Bilimler Araştırmaları Dergisi*, 3(1), 26–37.

14. Bagozi A., Bianchini D., de Antonellis V., Marini A., and Ragazzi D. (2017), Big Data Summarisation and Relevance Evaluation for Anomaly Detection in Cyber Physical Systems, In: H. Panetto, C. Debruyne, W. Gaaloul, M. Papazoglou, A. Paschke, C. A. Ardagna, R. Meersman (Eds.), *On the Move to Meaningful Internet Systems. OTM*

2017 Conferences. OTM 2017. Lecture Notes in Computer Science(), vol 10573. Cham: Springer, pp. 429-447.

15. Drath R. and Horch A. (2014), Industrie 4.0: Hit or hype? [Industry Forum], *IEEE Industrial Electronics Magazine*, 8(2), 56–58.

16. Kagermann H., Lukas WD., Wahlster W. (2011), Industrie 4.0: Mitdem Internet der Dinge auf demWegzur 4. *Industriellen Revolution, Meinung*. 13.

17. Alçın S., (2016), A New Theme For Productıon: Industry 4.0, *Journal of Life Economics*, 3(2), 19–30.

18. Heng S. (2014), Industry 4.0: Upgrading of Germany's Industrial Capabilities on the Horizon, *Deutsche Bank Research*, Current Issue, 1-16.

19. Özsoylu A. F. (2017), Endüstri 4.0, *Çukurova Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 21(1), 41–64.

20. Riordan A. O., Coady J., Newe T. and Dooly G., (2019, October), *Industry 4 .0 : Pillars for Smart Manufacturing – A Review,* Paper presented at the Industry 4.0 Summit, Hanoi, Vietnam.

21. Topkaya Ö. (2016), Dünyada Endüstriyel Robot Sektörü ve Çalışma Hayatına Etkileri, *Atatürk Üniversitesi İktisadi ve İdari Bilimler Dergisi*, 30(5), 1129–1144.

22. Shi Z., Xie Y., Xue W., Chen Y., Fu L., and Xu X. (2020), Smart factory in Industry 4.0, *Systems Research and Behavioral Science*, 37(4), 607–617.

23. Lasi H., Fettke P., Kemper H.-G., Feld T. and Hoffmann M., (2014), Industry 4.0, *Business & Information Systems Engineering*, 6(4), 239–242.

24. Osterrieder P., Budde L., and Friedli T. (2020), The smart factory as a key construct of industry 4.0: A systematic literature review, *International Journal of Production Economics*, 221, 107476.

25. Won J. Y. and Park M. J. (2020), Smart factory adoption in small and medium-sized enterprises: Empirical evidence of manufacturing industry in Korea, *Technological Forecasting and Social Change*, 157, 120117.

26. Wan J., Li J., Imran M. and Li D. (2019), A blockchain-based solution for enhancing security and privacy in smart factory, *IEEE Transactions on Industrial Informatics*, 15(6), 3652–3660.

27. Yoon S. C., Um J., Suh S. H., Stroud I., and Yoon J. S., (2019), Smart Factory Information Service Bus (SIBUS) for manufacturing application: requirement, architecture and implementation, *Journal of Intelligent Manufacturing*, 30(1), 363–382.

28. Radziwon A., Bilberg A., Bogers M., and Madsen E. S. (2014), The Smart Factory: Exploring Adaptive and Flexible Manufacturing Solutions, *Procedia Engineering*, 69, 1184–1190.

29. Napoleone A., Macchi M. and Pozzetti A. (2020), A review on the characteristics of cyber-physical systems for the future smart factories, *Journal of Manufacturing Systems*, 54, 305–335.

30. Rayes A. and Salam S. (2019), *Internet of Things From Hype to Reality: The Road to Digitization*, Cham, Switzerland: Springer Nature, 2.

31. Sidi M. S. A. and Zulhuda S. (2015), The Concept of Internet of Things and Its Challenges to Privacy, *South East Asia Journal of Contemporary Business, Economics and Law*, 8(4), 1–6.

32. Weber R. H. and Weber R. (2010), *Internet of Things: Legal Perspectives*, Berlin, Heidelberg: Springer, 1, 16-17.

33. Wortmann F. and Flüchter K. (2015), Internet of Things: Technology and Value Added, *Business and Information Systems Engineering*, 57(3), 221–224.

34. Greengard S. (2021), *The Internet of Things*, Cambridge: The MIT Press, 17, 53-54

35. Chander B., Pal S., De D., and Buyya R. (2022), Artificial Intelligence-based Internet of Things for Industry 5.0, In S. Pal, D. De, and R. Buyya, (Eds.) , *Artificial Intelligence-based Internet of Things Systems*, Cham: Springer International Publishing, pp. 3–45.

36. Yıldırım Y. (2019), Endüstri 4.0'a Kapsamlı Bir Bakış: 2011'den Bugüne, *Bilgi Dünyası*, 20(2), 217–249.

37. Yiğitbaşı H. Z. (2011, Ocak), *Nesnelerin İnterneti ve Makineden Makineye Kavramları İçin Kilit Öncül - IPv6*, Ulusal IPv6 Konferansında sunuldu, Ankara.

38. Chen B., Wan J., Shu L., Li P., Mukherjee M. and Yin B. (2017), Smart Factory of Industry 4.0: Key Technologies, Application Case, and Challenges, *IEEE Access*, 6, 6505–6519.

39. Zhong S., Zhong H., Huang X., Yang P., Shi J., Xie L. and Wang K. (2019), *Security and Privacy for Next-Generation Wireless Networks*, Cham: Springer International Publishing, 7-8.

40. Verma P. K., Verma R., Prakash A., Agrawal A., Naik K., Tripathi R., Alsabaan M., Khalifa T., Abdelkader T. and Abogharaf A. (2016), Machine-to-Machine (M2M) communications: A survey, *Journal of Network and Computer Applications*, 66, 83–105.

41. Marx L. R. K. and Kumar R. P. (2019), Locker Security System Using Matlab, *Lecture Notes in Networks and Systems*, 80, 280–288.

42. Polat H. and Oyucu S. (2017), Token-based authentication method for M2M platforms, *Turkish Journal of Electrical Engineering and Computer Science*, 25(4), 2956–2967.

43. Shen J., Zhou T., Liu X. and Chang Y. C. (2018), A novel Latin-square-based secret sharing for M2M communications, *IEEE Transactions on Industrial Informatics*, 14(8), 3659–3668.

44. Oyucu S. ve Polat H. (2017), M2M ve IoT Platformları Üzerinde Prototip Uygulama Geliştirme, *Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi*, 9(2), 11–20.

45. Chen H. C., You I., Weng C. E., Cheng C. H. and Huang Y. F. (2016), A security gateway application for End-to-End M2M communications, *Computer Standards & Interfaces*, 44, 85–93.

46. Elmas Ç. (2021), *Yapay Zeka Uygulamaları*, Ankara: Seçkin Yayıncılık, 25-26.

47. Nabiyev V. (2021), *Yapay Zeka: İnsan - Bilgisayar Etkileşimi*, Ankara: Seçkin Yayıncılık, 27.

48. Doğan O. ve Baloğlu N. (2020), Endüstri 4.0 Kavramsal Farkındalık Ölçeği, *KMÜ Sosyal ve Ekonomik Araştırmalar Dergisi*, 22(38), 58–81.

49. Nagar S. V., Chandrashekar A. C. and Suvarna M. (2020), Optimized Additive Manufacturing Technology Using Digital Twins and Cyber Physical Systems, In M. Auer, B. K. Ram (eds) *Cyber-physical Systems and Digital Twins. REV2019 2019. Lecture Notes in Networks and Systems*, 80. Cham: Springer, pp. 65-73.

50. Varshney S., Rohit S. S., Sunidhi, Jamkhandi A. G., Thanya D. and Parathodiyil M. (2020), Cyber-Physical Control and Virtual Instrumentation, In M. Auer, B. K. Ram (eds) *Cyber-physical Systems and Digital Twins. REV2019 2019. Lecture Notes in Networks and Systems*, 80. Cham: Springer, pp. 19-27.

51. Ertel W. (2017), *Introduction to Artificial Intelligence*. Cham: Springer International Publishing, 1.

52. Flasiński M. (2016), *Introduction to Artificial Intelligence*. Cham: Springer International Publishing, 222.

53. Akyıldız S. G. ve Akyıldız Y. (2020), Westworld Dizisinde Makine Öğrenmesi: Cinsiyet Eşitsizliğini Yeniden Üreten Yapay Zekâ Yaratımlar, *Journal of Turkish Studies*, 15(2), 999–1010.

54. Koşan M. A., Coşkun A. ve Karacan H. (2019), Yapay Zekâ Yöntemlerinde Entropi, *Journal of Information Systems and Management Research*, 1(1), 15–22.

55. Akın N. G. (2019), Makine Seçimi Probleminde Entropi - ROV ve CRITIC - ROV Yöntemlerinin Karşılaştırılması, *Dumlupınar Üniversitesi Sosyal Bilimler Dergisi*, (62), 20–39.

56. Shannon C. E. (1948), A Mathematical Theory of Communication, *Bell System Technical Journal*, 27(3), 379–423.

57. Oussous A., Benjelloun F. Z., Ait Lahcen A. and Belfkih S. (2018), Big Data technologies: A survey, *Journal of King Saud University - Computer and Information Sciences*, 30(4), 431–448.

58. Bhuvaneswari V. (2022), Role of Big Data Analytics in Industrial Revolution 4.0, In P. Kaliraj and T. Devi, (Eds.) , *Big Data Applications in Industry 4.0*, New York, NY, USA: Auerbach Publications, pp. 85–105.

59. Furht B. and Villanustre F. (2016), Introduction to Big Data, In B. Furht and F. Villanustre, (Eds.) , *Big Data Technologies and Applications*, Cham: Springer International Publishing, pp. 3–11.

60. Swaminathan K. and Palaniswami T. D. (2022), Breaking Technology Barriers in Diabetes and Industry 4.0, In P. Kaliraj and T. Devi, (Eds.) , *Big Data Applications in Industry 4.0*, New York: Auerbach Publications, pp. 71–83.

61. Olshannikova E., Ometov A., Koucheryavy Y. and Olsson T. (2016), Visualizing Big Data, In B. Furht and F. Villanustre, (Eds.) , *Big Data Technologies and Applications*, Cham: Springer, pp. 101–131.

62. Avcı A. F. ve Taşdemir Ş. (2019), Artırılmış ve sanal gerçeklik ile periyodik cetvel öğretimi, *Selcuk University Journal of Engineering Sciences*, 18(2), 68–83.

63. Özdemir O., Erbaş D. ve Yücesoy Özkan Ş. (2019), Özel Eğitimde Sanal Gerçeklik Uygulamaları, *Ankara University Faculty of Educational Sciences Journal of Special Education*, 20(2), 395–420.

64. Kim G. J. (2005), *Designing virtual reality systems: The structured approach*. London: Springer, 3.

65. Greengard S. (2019), *Virtual Reality*. Cambridge, United States: The MIT Press, 40.

66. de Souza Cardoso L. F., Mariano F. C. M. Q. and Zorzal E. R. (2020), A survey of industrial augmented reality, *Computers and Industrial Engineering*, 139, 106159.

67. Subramanian N. and Jeyaraj A. (2018), Recent security challenges in cloud computing, *Computers & Electrical Engineering*, 71, 28–42.

68. Srivastava P. and Khan R. (2018), A Review Paper on Cloud Computing, *International Journals of Advanced Research in Computer Science and Software Engineering*, 8(6), 17–20.

69. Luo G., Guo Y., Wang L., Li N. and Zou Y. (2021), Application of computer simulation and high-precision visual matching technology in green city garden landscape design, *Environmental Technology & Innovation*, 24, 101801.

70. Zhang L.,Zhou L., Ren L. and Laili Y. (2019), Modeling and simulation in intelligent manufacturing, *Computers in Industry*, 112, 103123.

71. Schluse M., Priggemeyer M., Atorf L. and Rossmann J. (2018), Experimentable Digital Twins-Streamlining Simulation-Based Systems Engineering for Industry 4.0, *IEEE Transactions on Industrial Informatics*, 14(4), 1722–1731.

72. Bandyopadhyay S. and Bhattacharya R. (2014), *Discrete and continuous simulation: Theory and practice*. Boca Raton: CRC Press, 2-3.

73. Zárate Ceballos H., Parra Amaris J. E., Jiménez Jiménez H., Romero Rincón D. A., Agudelo Rojas O. and Ortiz Triviño J. E. (2021), *Wireless Network Simulation*. Berkeley, CA: Apress, 4-5.

74. Carmona J. C., Giambiasi N. and Naamane A. (2004), Generalized Discrete Event Abstraction of Continuous Systems: Application to an Integrator, *Journal of Intelligent and Robotic Systems*, 41(1), 37–64.

75. Babulak E. and Wang M. (2010), Discrete Event Simulastion, In A. Goti, (Eds.) , *Discrete Event Simulations*, Rijeka, Croatia: Intech Open Science, pp. 1–10.

76. Kroll B., Schaffranek D., Schriegel S. and Niggemann O. (2014, January), *System modeling based on machine learning for anomaly detection and predictive maintenance in industrial plants,* Paper presented at the 19th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Barcelona, Spain.

77. Şen Ş. ve Yerlikaya T. (2013, Ocak), *ISO 27001 Kurumsal Bilgi Güvenliği Standardı,* Akademik Bilişim 2013 Konferansında sunuldu, Antalya.

78. Eminağaoğlu M. ve Gökşen Y. (2009), Bilgi Güvenliği Nedir, Ne Değildir? Türkiye'de Bilgi Güvenliği Sorunları ve Çözüm Önerileri, *Dokuz Eylül Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 11(4), 1–15.

79. Kizza J. M. (2017), *Guide to Computer Network Security*. Cham: Springer International Publishing, 43.

80. Campbell T. (2016), *Practical Information Security Management*. Berkeley, CA: Apress, 5-6.

81. Zdzikot T. (2022), Cyberspace and Cybersecurity, In K. Chałubińska-Jentkiewicz, F. Radoniewicz, and T. Zieliński, (Eds.) , *Cybersecurity in Poland*, Cham: Springer International Publishing, pp. 9–21.

82. Katsikeas S. Fysarakis K., Miaoudakis A., Van Bemten A., Askoxylakis I., Papaefstathiou I. and Plemenos A. (2017, September), *Lightweight & secure industrial IoT communications via the MQ telemetry transport protocol*, Paper presented at the IEEE Symposium on Computers and Communications, Heraklion, Greece, pp. 1193–1200.

83. Lee G., Epiphaniou G., Al-Khateeb H. and Maple C. (2019), Security and Privacy of Things: Regulatory Challenges and Gaps for the Secure Integration of Cyber-Physical Systems, *Advances in Intelligent Systems and Computing*, 797, 1–12.

84. Barki A., Bouabdallah A., Gharout S. and Traore J. (2016), M2M Security: Challenges and Solutions, *IEEE Communications Surveys and Tutorials*, 18(2), 1241–1254.

85. Pereira T., Barreto L. and Amaral A. (2017), Network and information security challenges within Industry 4.0 paradigm, *Procedia Manufacturing*, 13, 1253–1260.

86 Das T. K., Adepu S. and Zhou J. (2020), Anomaly detection in Industrial Control Systems using Logical Analysis of Data, *Computers & Security*, 96, 101935.

87. Kaur R. and Singh S. (2016), A survey of data mining and social network analysis based anomaly detection techniques, *Egyptian Informatics Journal*, 17(2), 199–216.

88. Nassif A. B., Talib M. A., Nasir Q. and Dakalbab F. M. (2021), Machine Learning for Anomaly Detection: A Systematic Review, *IEEE Access*, 9, 78658–78700.

89. Hayes M. A. and Capretz M. A. M. (2014, September), *Contextual anomaly detection in big sensor data*, Paper presented at the 2014 IEEE International Congress on Big Data, pp. 64–71, Anchorage, AK, USA.

90. Ahmed M. and Mahmood A. N. (2014, October), *Network traffic analysis based on collective anomaly detection*, Paper presented at the 2014 9th IEEE Conference on Industrial Electronics and Applications, ICIEA 2014, pp. 1141–1146, Hangzhou, China.

91. Ahmed M. (2018), Collective Anomaly Detection Techniques for Network Traffic Analysis, *Annals of Data Science*, 5(4), 497–512.

92. Ahmed M. and Pathan A. S. K. (2019), Investigating Deep Learning for Collective Anomaly Detection - An Experimental Study, In S. M. Thampi, S. Madria, G. Wang, D. B. Rawat, and J. M. A. Calero, (Eds.) , *Security in Computing and Communications*, 969, Singapore: Springer, pp. 211–219.

93. Batarseh F. A. (2018), Anomaly Detection, In L. A. Schintler and C. L. McNeely, (Eds.), *Encyclopedia of Big Data*, Cham: Springer International Publishing, pp. 25–26.

94. Nguyen V. Q., van Ma L. and Kim J. (2018), LSTM-based Anomaly Detection on Big Data for Smart Factory Monitoring, *Journal of Digital Contents Society*, 19(4), 789–799.

95. Hwang G., Kang S., Dweekat A. J., Park J. and Chang T. W. (2018), An IoT data anomaly response model for smart factory performance measurement, International *Journal of Industrial Engineering : Theory Applications and Practice*, 25(5), 702–718.

96. Maier A., Schriegel S. and Niggemann O. (2017), Big Data and Machine Learning for the Smart Factory—Solutions for Condition Monitoring, Diagnosis and Optimization, In S. Jeschke, C. Brecher, H. Song, D. Rawat (Eds.) *Industrial Internet of Things. Springer Series in Wireless Technology*. Cham: Springer, pp. 473-485.

97. Yang H., Xie F. and Lu Y. (2006), Clustering and Classification Based Anomaly Detection, In L. Wang, L. Jiao, G. Shi, X. Li, and J. Liu, (Eds.) , *Fuzzy Systems and Knowledge Discovery. FSKD 2006. Lecture Notes in Computer Science*, Berlin, Heidelberg: Springer, pp. 1082–1091.

98. Chandola V., Banerjee A. and Kumar V. (2009), Anomaly Detection : A Survey, *ACM Computing Surveys*, 41(3), 1–58.

99. Zhang W., Yang Q. and Geng Y. (2009, January), *A survey of anomaly detection methods in networks*, Paper presented at the 1st International Symposium on Computer Network and Multimedia Technology, Wuhan, China, pp. 1–3.

100. Ahmed M., Naser Mahmood A. and Hu J. (2016), A survey of network anomaly detection techniques, *Journal of Network and Computer Applications*, 60, 19–31.

101. Bellala G., Marwah M., Shah A., Arlitt M. and Bash C. (2012, November), *A finite state machine-based characterization of building entities for monitoring and control*, Paper presented at the 4th ACM Workshop on Embedded Systems for Energy Efficiency in Buildings, pp. 153–160, Ontario, Canada.

102. Ju H., Lee D., Hwang J., Namkung J. and Yu H. (2020), PUMAD: PU Metric learning for anomaly detection, *Information Sciences*, 523, 167–183.

103. Baek S., Kwon D., Suh S. C., Kim H., Kim I. and Kim J. (2021), Clustering-based label estimation for network anomaly detection, *Digital Communications and Networks*, 7(1), 37–44.

104. Wu H., Shen Y., Xiao X., Hecker A. and Fitzek F. H. P. (2021, December), In-Network Processing Acoustic Data for Anomaly Detection in Smart Factory, Paper presented at the IEEE Gobal Communication Conference, Madrid, Spain.

105. Hasan M., Islam M. M., Zarif M. I. I. and Hashem M. M. A. (2019), Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches, *Internet of Things*, 7, 100059.

106. Hsieh R. J., Chou J. and Ho C. H. (2019, January), *Unsupervised online anomaly detection on multivariate sensing time series data for smart manufacturing*, Paper presented at the 2019 IEEE 12th Conference on Service-Oriented Computing and Applications, pp. 90–97, Kaohsiung, Taiwan.

107. Sarafijanovic-Djukic N. and Davis J. (2019), Fast Distance-Based Anomaly Detection in Images Using an Inception-Like Autoencoder, in P. K. Novak, T. Šmuc, and S. Džeroski (Eds.), *Discovery Science*, Cham: Springer, pp. 493–508.

108. Teng M. (2010, January), *Anomaly detection on time series*, Paper presented at the 2010 IEEE International Conference on Progress in Informatics and Computing, Shangai, China, pp. 603–608.

109. Kang M. (2019), Machine Learning: Anomaly Detection, In M. G. Pecht and M. Kang, (Eds.), *Prognostics and Health Management of Electronics: Fundamentals, Machine Learning, and the Internet of Things*, Chichester, UK: Wiley-IEEE Press, pp. 131–162.

110. Kumpulainen P., Kylväjä M. and Hätönen K. (2009, September), *Importance Of Scalıng In Unsupervısed Dıstance-Based Anomaly Detectıon*, Paper presented at the XIX IMEKO World Congress Fundamental and Applied Metrology, Lisbon, Portugal, pp. 2411–2416.

111. Kiss I., Genge B., Haller P. and Sebestyen G. (2014, September), *Data clustering-based anomaly detection in industrial control systems*, Paper presented at the 2014

IEEE 10th International Conference on Intelligent Computer Communication and Processing, Cluj-Napoca, Cluj, Romania, pp. 275–281.

112. Syarif I., Prugel-Bennett A. and Wills G. (2012), Unsupervised Clustering Approach for Network Anomaly Detection, in R. Benlamri (Eds.), *Networked Digital Technologies. NDT 2012. Communications in Computer and Information Science*, Berlin, Heidelberg: Springer, pp. 135–145.

113. Li J., Izakian H., Pedrycz W. and Jamal I. (2021), Clustering-based anomaly detection in multivariate time series data, *Applied Soft Computing*, 100, 106919.

114. Loo C. E., Ng M. Y., Leckie C. and Palaniswami M. (2006), Intrusion detection for routing attacks in sensor networks, *International Journal of Distributed Sensor Networks*, 2(4), 313–332.

115. Klerx T., Anderka M., Buning H. K. and Priesterjahn S. (2014, December), *Model-Based Anomaly Detection for Discrete Event Systems*, Paper presented at the International Conference on Tools with Artificial Intelligence, ICTAI, Limassol, Cyprus, pp. 665–672.

116. Simon D. L. and Rinehart A. W. (2014, June), *A Model-Based Anomaly Detection Approach for Analyzing Streaming Aircraft Engine Measurement Data*, Paper presented at the ASME Turbo Expo 2014: Turbine Technical Conference and Exposition, Düsseldorf, Germany, pp. 1–11.

117. Li P. and Niggemann O. (2021), A Nonconvex Archetypal Analysis for One-Class Classification Based Anomaly Detection in Cyber-Physical Systems, *IEEE Transactions on Industrial Informatics*, 17(9), 6429–6437.

118. Sarkar D., Bali R. and Sharma T. (2018), *Practical Machine Learning with Python*. Berkeley, CA: Apress, 3, 64.

119. Mishra S. and Tyagi A. K. (2022), The Role of Machine Learning Techniques in Internet of Things-Based Cloud Applications, In S. Pal, D. De, and R. Buyya, (Eds.), *Artificial Intelligence-based Internet of Things Systems*, Cham: Springer International Publishing, pp. 105–135.

120. Jiang T., Gradus J. L. and Rosellini A. J. (2020), Supervised Machine Learning: A Brief Primer, *Behavior Therapy*, 51(5), 675–687.

121. Singh A., Thakur N. and Sharma A. (2016, March), *A review of supervised machine learning algorithms*, Paper presented at the 3rd International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, pp. 1310–1315.

122. Kapitanova K. and Son S. H. (2013), Machine Learning Basics, In F. Hu and Q. Hao, (Eds.), *Intelligent Sensor Networks: The Integration of Sensor Networks, Signal Processing and Machine Learning*, Boca Raton, USA: CRC Press, pp. 3–27.

123. Jung A. (2022), *Machine Learning: The Basics*. Singapore: Springer, 12, 14.

124. Rebala G., Ravi A. and Churiwala S. (2019), *An Introduction to Machine Learning*. Cham: Springer International Publishing, 1.

125. Şeker Ş. E. (2016), *Weka ile Veri Madenciliği*. İstanbul: Bilgisayar Kavramları Yayınları, 9, 18-19.

126. Akçetin E. ve Çelik U. (2014), İstenmeyen Elektronik Posta (Spam) Tespitinde Karar Ağacı Algoritmalarının Performans Kıyaslaması, *İnternet Uygulamaları ve Yönetimi Dergisi*, 5(2), pp. 43–56.

127. İşçimen B., Kutlu Y., Reyhaniye A. N. ve Turan C. (2014, Nisan), Balık tanınmasında görüntü analiz yöntemleri, Paper presented at the IEEE 22nd Signal Processing and Communications Applications Conference (SIU 2014), Trabzon, pp. 1411–1414.

128. Nizam H. ve Akın S. S. (2014, Kasım), Sosyal Medyada Makine Öğrenmesi ile Duygu Analizinde Dengeli ve Dengesiz Veri Setlerinin Performanslarının Karşılaştırılması, XIX. Türkiye'de İnternet Konferansında sunulmuştur (inet-tr'14), İzmir, pp. 77-84.

129. Han J. and Kamber M. (2006), *Data Mining Concepts and Techniques. 2nd Edition.* San Francisco, USA: Morgan Kaufmann Publishers, 311-313, 328-329.

130. Hand D., Mannila H. and Smyth P. (2001), *Principles of Data Mining*, Cambridge: MIT Press, 211-213.

131. Patil T. R. and Sherekar M. S. S. (2013), Performance Analysis of Naive Bayes and J48 Classification Algorithm for Data Classification, *International Journal Of Computer Science And Applications*, 6(2), 256-261.

132. Aydemir E. (2019), *Weka ile Yapay Zeka*. Ankara: Seçkin Yayıncılık, 26-27, 33.

133. Staub S., Karaman E., Kaya S., Karapnar H. and Güven E. (2015), Artificial Neural Network and Agility, *Procedia-Social and Behavioral Sciences*, 195, 1477–1485.

134. Yakut E., Elmas B. ve Yavuz S. (2014), Yapay Sinir Ağları ve Destek Vektör Makineleri Yöntemleriyle Borsa Endeksi Tahmini, *Süleyman Demirel Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 19(1), pp. 139–157.

135. Arı A. ve Berberler M. E. (2017), Yapay Sinir Ağları ile Tahmin ve Sınıflandırma Problemlerinin Çözümü İçin Arayüz Tasarımı, *Acta Infologica*, 1(2), pp. 55–73.

136. Brownlee J. (2020), *Machine Learning Mastery With Weka*, Melbourne: Independently Published, 115, 119-120.

137. Mansournia M. A., Geroldinger A., Greenland S. and Heinze G. (2018), Separation in Logistic Regression: Causes, Consequences, and Control, *American Journal of Epidemiology*, 187(4), 864–870.

138. Jain M., Narayan S., Balaji P., Bharath K. P., Bhowmick A., Karthik R. and Muthu R. K. (2018, March), Paper presented at the *Speech Emotion Recognition using*

Support Vector Machine, Presented at the International Conference on Informatics Computing in Engineering Systems (ICICES), Chennai, India.

139. Feizizadeh B., Roodposhti M. S., Blaschke T. and Aryal J. (2017), Comparing GIS-based support vector machine kernel functions for landslide susceptibility mapping, *Arabian Journal of Geosciences*, 10(5), 1–13.

140. Alpaydın E. (2018), *Yapay Öğrenme*. İstanbul: Boğaziçi Üniversitesi Yayınevi, 153.

141. Aksu M. ve Karaman E. (2017), Karar Ağaçları ile Bir Web Sitesinde Link Analizi ve Tespiti, *Acta Infologica*, 1(2), 84–91.

142. Tanha J., van Someren M. and Afsarmanesh H. (2017), Semi-supervised self-training for decision tree classifiers, *International Journal of Machine Learning and Cybernetics*, 8(1), 355–370.

143. Belgiu M. and Drăgu L. (2016), Random forest in remote sensing: A review of applications and future directions, *ISPRS Journal of Photogrammetry and Remote Sensing*, 114, 24–31.

144. Holzinger A., Malle B., Kieseberg P., Roth P. M., Müller H., Reihs R. and Zatloukal K. (2017), Machine learning and knowledge extraction in digital pathology needs an integrative approach, In A. Holzinger, R. Goebel, M. Ferri, V. Palade (eds.) *Towards Integrative Machine Learning and Knowledge Extraction. Lecture Notes in Computer Science,* Cham: Springer, pp. 13-50.

145. Breiman L. (2001), Random Forests, *Machine Learning*, 45(1), 5–32.

146. Harefa J., Alexander A. and Pratiwi M. (2017), Comparison Classifier: Support Vector Machine (SVM) and K-Nearest Neighbor (K-NN) In Digital Mammogram Images, *Jurnal Informatika dan Sistem Informasi*, 2(2), 35–40.

147. Indriani O. R., Kusuma E. J., Sari C. A., Rachmawanto E. H. and Setiadi D. R. I. M. (2017, November), *Tomatoes classification using K-NN based on GLCM and HSV color space,* Paper presented at the 2017 International Conference on Innovative and Creative Information Technology: Computational Intelligence and IoT, ICITech, Salatiga, Indonesia pp. 1–6.

148. Jadhav S. D. and Channe H. P. (2013), Comparative Study of K-NN, Naive Bayes and Decision Tree Classification Techniques, *International Journal of Science and Research (IJSR)*, 5, 1842–1845.

149. Touzani S., Granderson J. and Fernandes S. (2018), Gradient boosting machine for modeling the energy consumption of commercial buildings, *Energy and Buildings*, 158, 1533–1543.

150. Natekin A. and Knoll A. (2013), Gradient boosting machines, a tutorial, *Frontiers in Neurorobotics*, 7, 1–21.

151. Ayyadevara V. K. (2018), *Pro Machine Learning Algorithms*. Berkeley, CA: Apress, 117-118.

152. İnternet: Malohlava M. and Candel A. Gradient Boosting Machine with H2O, *h2o.ai Resources*, vol. 7, URL: https://h2o.ai/resources/booklet/gradient-boosting-machine-with-h2o/, Son Erişim Tarihi: 24.04.2022.

153. Konstantinov A. V. and Utkin L. V. (2021), Interpretable machine learning with an ensemble of gradient boosting machines, *Knowledge-Based Systems*, 222, 106993.

154. Fan J., Ma X., Wu L., Zhang F., Yu X. and Zeng W. (2019), Light Gradient Boosting Machine: An efficient soft computing model for estimating daily reference evapotranspiration with local and external meteorological data, *Agricultural Water Management*, 225, 105758.

155. Li F., Zhang L., Chen B., Gao D., Cheng Y., Zhang X., Yang Y., Gao K., Huang Z. and Peng J. (2018, November), *A Light Gradient Boosting Machine for Remainning Useful Life Estimation of Aircraft Engines*, Paper presented at the IEEE Conference on Intelligent Transportation Systems, Proceedings (ITSC), Maui, HI, USA, pp. 3562–3567.

156. Alzamzami F., Hoda M. and el Saddik A. (2020), Light Gradient Boosting Machine for General Sentiment Classification on Short Texts: A Comparative Evaluation, *IEEE Access*, 8, 101840–101858.

157. jo Chun P., Izumi S. and Yamane T. (2021), Automatic detection method of cracks from concrete surface imagery using two-step light gradient boosting machine, *Computer-Aided Civil and Infrastructure Engineering*, 36(1), 61–72.

158. İnternet: LightGBM Documentation, *Microsoft Features*, URL: https://lightgbm.readthedocs.io/en/latest/Features.html, Son Erişim Tarihi: 24.04.2022.

159. Gijsbers P., LeDell E., Thomas J., Poirier S., Bischl B. and Vanschoren J. (2019, June), *An Open Source AutoML Benchmark*, Paper presented at the 6th ICML Workshop on Automated Machine Learning, Long Beach, California, USA, pp. 1-8.

160. Zöller M. A. and Huber M. F. (2021), Benchmark and Survey of Automated Machine Learning Frameworks, *Journal of Artificial Intelligence Research*, 70, 409–472.

161. He X., Zhao K. and Chu X. (2021), AutoML: A survey of the state-of-the-art, *Knowledge-Based Systems*, 212, 106622.

162. Zhang S., Zhang C. and Yang Q. (2003), Data preparation for data mining, *Applied Artificial Intelligence*, 17(5–6), 375–381.

163. Zheng A. and Casari A. (2018), *Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists*. USA: O'Reilly Media, 9-10.

164. Li Y. and Li T. (2018), Feature Selection and Evaluation, In G. Dong and H. Liu, (Eds.), *Feature Engineering for Machine Learning and Data Analytics*, Boca Raton, USA: CRC Press, pp. 191–220.

165. Pechenizkiy M., Puuronen S. and Tsymbal A. (2003), Feature Extraction for Classification in The Data Mining Process, *International Journal Information Theories & Applications*, 10, 271–278.

166. Smith M. G. and Bull L. (2005), Genetic Programming with a Genetic Algorithm for Feature Construction and Selection, *Genetic Programming and Evolvable Machines*, 6(3), 265–281.

167. Peng D., Dong X., Real E., Tan M., Lu Y., Liu H., Bender G., Kraft A., Liang C. and Le Q. V. (2020, December), *PyGlove: Symbolic Programming for Automated Machine Learning*, Paper presented at the 34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

168. Barboza F., Kimura H. and Altman E. (2017), Machine learning models and bankruptcy prediction, *Expert Systems with Applications*, 83, 405–417.

169. Luu S. T., Nguyen H. P., van Nguyen K. and Luu-Thuy Nguyen N. (2020, October), *Comparison between Traditional Machine Learning Models and Neural Network Models for Vietnamese Hate Speech Detection*, Paper presented at the 2020 RIVF International Conference on Computing and Communication Technologies (RIVF), Ho Chi Minh City, Vietnam, pp. 1-6.

170. Yang L. and Shami A. (2020), On hyperparameter optimization of machine learning algorithms: Theory and practice, Neurocomputing, 415, pp. 295–316.

171. Truong A., Walters A., Goodsitt J., Hines K., Bruss C. B. and Farivar R. (2019, November), *Towards automated machine learning: Evaluation and comparison of AutoML approaches and tools*, Paper presented at the International Conference on Tools with Artificial Intelligence (ICTAI), Portland, OR, USA, pp. 1471–1479.

172. Jin H., Song Q. and Hu X. (2019, August), *Auto-Keras: An Efficient Neural Architecture Search System*, Paper presented at the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage AK USA, pp. 1946-1956.

173. Feurer M., Klein A., Eggensperger K., Springenberg J. T., Blum M. and Hutter F. (2019), Auto-sklearn: Efficient and Robust Automated Machine Learning, In F. Hutter, L. Kotthoff, and J. Vanschoren, (Eds.), *Automated Machine Learning: Methods, Systems, Challenges*, Cham: Springer, pp. 113–134.

174. Gain U. and Hotti V. (2021), Low-code AutoML-augmented Data Pipeline – A Review and Experiments, *Journal of Physics: Conference Series*, 1828(1), 012015.

175. Özdemir Ş. ve Örslü S. (2019), Makine Öğrenmesinde Yeni Bir Bakış Açısı : Otomatik Makine Öğrenmesi (AutoML), *Journal of Information Systems and Management Research*, 1(1), 23–29.

176. Ferreira L., Pilastri A., Martins C. M., Pires P. M. and Cortez P. (2021, July), *A Comparison of AutoML Tools for Machine Learning, Deep Learning and XGBoost*, Paper presented at the International Joint Conference on Neural Networks, Shenzhen, China, pp. 1-8.

177. BudjačR., Nikmon M., Schreiber P., Zahradníková B. and Janáčová D. (2019), Automated Machine Learning Overview, *Research Papers Faculty of Materials Science and Technology Slovak University of Technology*, 27(45), 107–112.

178. Fabris F. and Freitas A. A. (2019), Analysing the Overfit of the Auto-sklearn Automated Machine Learning Tool, In G. Nicosia, P. Pardalos, R. Umeton, G. Giuffrida, V. Sciacca (eds.) *Machine Learning, Optimization, and Data Science*, Cham: Springer, pp. 508-520.

179. van Eeden W. A., Luo C., van Hemert A. M., Carlier I. V. E., Penninx B. W., Wardenaar K. J., Hoos H. and Giltaya E. J. (2021), Predicting the 9-year course of mood and anxiety disorders with automated machine learning: A comparison between auto-sklearn, naïve Bayes classifier, and traditional logistic regression, *Psychiatry Research*, 299, 113823.

180. Munjal A., Khandia R., and Gautam B., (2020), A Machıne Learnıng Approach For Selectıon Of Polycystıc Ovarıan Syndrome (Pcos) Attrıbutes And Comparıng Dıfferent Classıfier Performance Wıth The Help Of Wekaand Pycaret, *Internatıonal Journal Of Scıentıfic Research*, 9(12), 59–63.

181. İnternet: Moez A. PyCaret: An open source, low-code machine learning library in Python, *PyCaret*, URL: https://pycaret.readthedocs.io/en/latest/index.html, Son Erişim Tarihi: 16.02.2022.

182. Avendano J. C., Otero L. D. and Otero C. (2021, April), *Application of Statistical Machine Learning Algorithms for Classification of Bridge Deformation Data Sets*, Paper presented at the 15th Annual IEEE International Systems Conference, Vancouver, BC, Canada, pp. 1-7.

183. Ramani T., Sirihsa Madhuri T. and Jayasri P. (2021), Bayesian Optimisation Search (BOS) Based Future Sales Prediction, *A Journal of Composition Theory*, 14(12), 50-56.

184. Kurian J. J., Dix M., Amihai I., Ceusters G. and Prabhune A. (2021, August), *BOAT: A Bayesian Optimization AutoML Time-series Framework for Industrial Applications*, Paper presented ay the IEEE 7th International Conference on Big Data Computing Service and Applications, Oxford, United Kingdom, pp. 17–24.

185. Olson R. S., Bartley N., Urbanowicz R. J. and Moore J. H. (2016, July), *Evaluation of a tree-based pipeline optimization tool for automating data science*, Paper presented at the 2016 Genetic and Evolutionary Computation Conference, Denver, Colorado, USA, pp. 485–492.

186. Ledell E. and Poirier S. (2020, July), *H2O AutoML: Scalable Automatic Machine Learning*, Paper presented at the 7th ICML Workshop on Automated Machine Learning, Vienna, Austria, pp. 1–16.

187. Zeng Y. and Zhang J. (2020), A machine learning model for detecting invasive ductal carcinoma with Google Cloud AutoML Vision, *Computers in Biology and Medicine*, 122, 103861.

188. Chen Y.-W., Song Q. and Hu X., (2021), Techniques for Automated Machine Learning, *ACM SIGKDD Explorations Newsletter*, 22(2), 35–50.

189. Zimmer L., Lindauer M. and Hutter F. (2021), Auto-Pytorch: Multi-Fidelity MetaLearning for Efficient and Robust AutoDL, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(9), 3079–3090.

190. Kotthoff L., Thornton C., Hoos H. H., Hutter F. and Leyton-Brown K. (2019), Auto-WEKA: Automatic Model Selection and Hyperparameter Optimization in WEKA, In F. Hutter, L. Kotthoff, and J. Vanschoren, (Eds.), *Automated Machine Learning: Methods, Systems, Challenges*, Cham, Switzerland: Springer, pp. 81–95.

191. Hranisavljevic N., Niggemann O. and Maier A. (2016, October), *A Novel Anomaly Detection Algorithm for Hybrid Production Systems based on Deep Learning and Timed Automata*, Paper presented at the The 27th International Workshop on Principles of Diagnosis (DX-2016At), Denver, Colorado, USA.

192. von Birgelen A. and Niggemann O. (2017, September), *Using self-organizing maps to learn hybrid timed automata in absence of discrete events*, Paper presented at the IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Limassol, Cyprus, pp. 1–8.

193. Seo CB., Lee G., Lee Y. and Seo SH. (2021), Echo-Guard: Acoustic-Based Anomaly Detection System for Smart Manufacturing Environments, In H. Kim (Eds.) *Information Security Applications. WISA 2021. Lecture Notes in Computer Science*, vol 13009, Cham: Springer, pp. 64-75.

194. Lam J. and Abbas R. (2020), Machine Learning based Anomaly Detection for 5G Networks, *arXiv - CS - Cryptography and Security*, abs/2003.03474, 1-12.

195. Wu Y., Dai H. N. and Tang H. (Baskıda), Graph Neural Networks for Anomaly Detection in Industrial Internet of Things, *IEEE Internet of Things Journal*, 1-18, doi: 10.1109/JIOT.2021.3094295.

196. Pahl M.-O. and Aubet F.-X. (2018, November), *All Eyes on You: Distributed Multi-Dimensional IoT Microservice Anomaly Detection*, Paper presented at the 14th International Conference on Network and Service Management (CNSM), Rome, Italy, pp. 72-80.

197. Wang R., Nie K., Wang T., Yang Y. and Long B. (2020, January), *Deep Learning for Anomaly Detection*, Paper presented at the 13th International Conference on Web Search and Data Mining, Houston TX, USA, pp. 894–896.

198. Hranisavljevic N., Maier A. and Niggemann O. (2020), Discretization of hybrid CPPS data into timed automaton using restricted Boltzmann machines, *Engineering Applications of Artificial Intelligence*, 95, 103826.

199. Kim K. H., Shim S., Lim Y., Jeon J., Choi J., Kim B. and Yoon A. S. (2020, April), *RaPP: Novelty Detection with Reconstruction along Projection Pathway*, Paper presented at the 8th International Conference on Learning Representations (ICLR), New York, USA.

200. Kim D., Cha J., Oh S. and Jeong J. (2021, January), *AnoGAN-Based Anomaly Filtering for Intelligent Edge Device in Smart Factory*, Paper presetented at the 15th International Conference on Ubiquitous Information Management and Communication (IMCIM 2021), Seoul, Korea (South), pp: 1-6.

201. Tien C. W., Huang T. Y., Chen P. C. and Wang J. H. (2021), Using Autoencoders for Anomaly Detection and Transfer Learning in IoT, *Computers 2021*, 10(7), 88-102.

202. Huong T. T., Bac T. P., Long D. M., Luong T. D., Dan N. M., Quang L. A., Cong L. T., Thang B. D. and Tran K. P. (2021), Detecting cyberattacks using anomaly detection in industrial control systems: A Federated Learning approach, *Computers in Industry*, 132, 103509.

203. Savic M., Lukic M., Danilovic D., Bodroski Z., Bajovic D., Mezei I, Vukobratovic D., Skrbic S. and Jakovetic D. (2021), Deep Learning Anomaly Detection for Cellular IoT with Applications in Smart Logistics, *IEEE Access*, 9, 59406–59419.

204. P. Stahmann and B. Rieger (2021, January), *Requirements identification for real-time anomaly detection in Industrie 4.0 machine groups: A structured literature review*, Paper presented at the Annual Hawaii International Conference on System Sciences, Hawaii, USA, pp. 5738–5747.

205. Souza M. L. H., da Costa C. A., Ramos G. O. and Righi R. R. (2021), A feature identification method to explain anomalies in condition monitoring, *Computers in Industry*, 133, 103528.

206. Andy S., Rahardjo B. and Hanindhito B. (2017, September), *Attack scenarios and security analysis of MQTT communication protocol in IoT system*, Paper presented at the International Conference on Electrical Engineering, Computer Science and Informatics (EECSI), Yogyakarta, Indonesia, pp. 1-6.

207. İnternet: MQTT, Frequently Asked Questions, *MQTT*, URL: https://mqtt.org/faq/, Son Erişim Tarihi: 14.02.2022.

208. İnternet: Versatile Production System, *Kaggle Your Home Data Sci.*, URL: https://www.kaggle.com/inIT-OWL/versatileproductionsystem, Son Erişim Tarihi: 20.04.2020.

209. Yılmaz Redüktör (2011), *Mekanik Uygulamalar: Temel Formüller*, İstanbul: Yılmaz Redüktör, 9.

210. İnternet: Pompa Gücü Nasıl Hesaplanır?, *Genel Mekanik Ansiklopedisi*, URL: https://genelmekanik.com/index.php/2020/08/17/pompa-gucu-nasil-hesaplanir/, Son Erişim Tarihi: 15.02.2022.

GAZİLİ OLMAK AYRICALIKTIR…