

Fabrikalardaki Otonom Taşıyıcı Araçlar için Veri Odaklı Anomali Tespiti

Özlem Örnek

**YÜKSEK LİSANS TEZİ**

Bilgisayar Mühendisliği Anabilim Dalı

Aralık 2020

Data Driven Anomaly Detection for Autonomous Transport Vehicles in Factories

Özlem Örnek

**MASTER OF SCIENCE THESIS**

Department of Computer Engineering

December 2020

Fabrikalardaki Otonom Taşıyıcı Araçlar için Veri Odaklı Anomali Tespiti

Özlem Örnek

Eskişehir Osmangazi Üniversitesi  
Fen Bilimleri Enstitüsü  
Lisansüstü Yönetmeliği Uyarınca  
Bilgisayar Mühendisliği Anabilim Dalı  
Bilgisayar Yazılımı Bilim Dalında  
YÜKSEK LİSANS TEZİ  
Olarak Hazırlanmıştır

Danışman: Doç. Dr. Ahmet Yazıcı

Bu tez, Türkiye Bilimsel ve Teknolojik Araştırma Kurumu'nun (TUBİTAK) 116E731  
nolu projesi tarafından desteklenmiştir

Aralık 2020

## ETİK BEYAN

Eskişehir Osmangazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kılavuzuna göre, Doç. Dr. Ahmet Yazıcı danışmanlığında hazırlamış olduğum “Fabrikalardaki Otonom Taşıyıcı Araçlar için Veri Odaklı Anomali Tespiti” başlıklı YÜKSEK LİSANS tezimin özgün bir çalışma olduğunu; tez çalışmamın tüm aşamalarında bilimsel etik ilke ve kurallara uygun davrandığımı; tezimde verdiğim bilgileri, verileri akademik ve bilimsel etik ilke ve kurallara uygun olarak elde ettiğimi; tez çalışmamda yararlandığım eserlerin tümüne atıf yaptığımı ve kaynak gösterdiğimi ve bilgi, belge ve sonuçları bilimsel etik ilke ve kurallara göre sunduğumu beyan ederim. 14/12/2020

Özlem Örnek

İmza

## ÖZET

Endüstri 4.0 ile beraber akıllı fabrikaları gerçeklemek üzere otonom sistemlerin geliştirilmesi öngörülmektedir. Bu sistemlerin kontrolünün, izlenmesinin veya olası anormal durum tespitlerinin insan müdahalesi olmaksızın veri üzerinden yapılması beklenmektedir. Endüstri 4.0'ın gerçekleşmesinde kritik öneme sahip teknolojilerden birisi de otonom robotlardır. Bu bağlamda akıllı fabrikalarda iç lojistik amaçlı kullanılan ve taşıma görevlerini gerçekleştiren otonom taşıyıcı araç (OTA)'ların izlenmesi önemlidir. OTA'ların sorunsuz olarak çalışması hem sistem güvenliği hem de verimlilik için gereklidir. Bu nedenle OTA'lardan oluşan sistemlerde oluşabilecek anormal durumların tespit edilmesi önemlidir.

Bu çalışmada, OTA sistemlerinde veri üzerinden anormal durum tespitine yönelik iki problem ele alınmaktadır. İlk olarak fabrika seviyesinde OTA'ların oluşturduğu trafikteki anormal durumların tespit edilmesine yönelik çözüm geliştirilmektedir. Burada anormal durumların denetimli öğrenme ile öğrenilmesi ve sonrasında ise anomali tespiti yapılmaktadır. Literatürdeki yöntemlerinden karar ağacı ve bulanık mantık kullanılarak OTA'ların oluşturduğu trafikteki anormal durum tespit edilmektedir. İkinci olarak ise OTA konumlandırma sisteminde kullanılan konum hesaplama yaklaşımlarında anormal durum tespiti yapılmaktadır. Bu yaklaşımda herhangi bir ortama ait OTA'da bulunan üç farklı konum hesaplama yönteminden elde edilen veriler "Üst Güven Sınırı (Upper Confidence Bound, UCB)" algoritması ile öncelikle öğrenilmekte, sonrasında ise buradaki ödüldeki olası değişimler İstatistiksel Proses Kontrol (İPK) ile tespit edilmektedir. Önerilen her iki anormal durum tespit yaklaşımı, ESOGÜ Akıllı Fabrika ve Robotik Laboratuvarında (Intelligent Factory and Robotic Laboratory, IFARLAB) kurgulanan farklı deney ortamlarında oluşan veriler ile test edilmiştir.

**Anahtar Kelimeler:** Anomali tespiti, Otonom robot, Endüstri 4.0, Akıllı fabrika, Gözetimli öğrenme, Pekiştirmeli öğrenme.

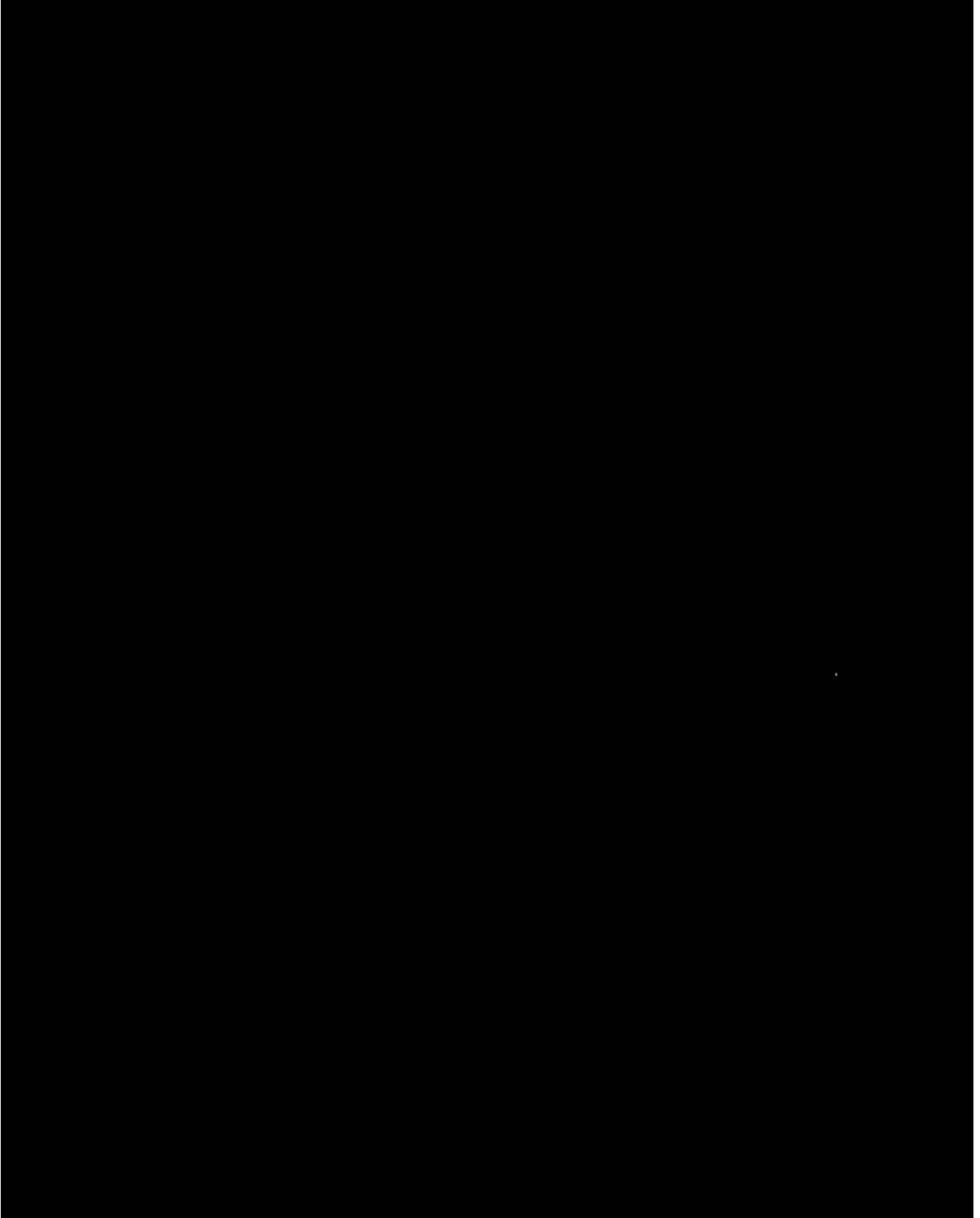
## SUMMARY

Autonomous systems will be developed to realize smart factories in the context of Industry 4.0. It is important to monitor and control these systems over data and detect possible anomalies without human intervention. Autonomous robots are one of the critical technologies in the realization of Industry 4.0. In this context, monitoring of Autonomous Transport Vehicle (ATV)'s that used for internal logistics to perform transport tasks is important. The correct operation of ATVs is necessary for both system safety and efficiency. For this reason, it is important to detect abnormal situations that may occur in systems consisting of ATVs.

In this study, two problems for detecting anomalies over data in ATV systems are discussed. Firstly, a solution is developed to detect abnormal situations in the traffic generated by ATVs at the factory level. Here, abnormal situations are learned by supervised learning and then anomaly detection is made. The decision tree and fuzzy logic are used to determine the abnormal situation in the traffic generated by ATVs. Secondly, abnormal situation detection is performed in the positioning system of ATV. In this approach, the data obtained from three different position calculation methods learned with the Upper Confidence Bound (UCB) algorithm, and then the possible changes in the reward here are determined by Statistical Process Control (SPC). Both proposed anomaly detection approaches have been tested with the data generated in different experimental environments set up in ESOGÜ Intelligent Factory and Robotic Laboratory (IFARLAB).

**Keywords:** Anomaly detection, Autonomous robot, Industry 4.0, Smart factory, Supervised learning, Reinforcement learning.

## TEŞEKKÜR



# İÇİNDEKİLER

	<u>Sayfa</u>
<b>ÖZET</b> .....	<b>v</b>
<b>SUMMARY</b> .....	<b>vi</b>
<b>TEŞEKKÜR</b> .....	<b>vii</b>
<b>İÇİNDEKİLER</b> .....	<b>viii</b>
<b>ŞEKİLLER DİZİNİ</b> .....	<b>ix</b>
<b>ÇİZELGELER DİZİNİ</b> .....	<b>x</b>
<b>SİMGELER VE KISALTMALAR DİZİNİ</b> .....	<b>xi</b>
<b>1. GİRİŞ VE AMAÇ</b> .....	<b>1</b>
<b>2. LİTERATÜR ARAŞTIRMASI</b> .....	<b>4</b>
2.1. Anomali Tespiti .....	<b>4</b>
2.1.1. Anomali tespiti yöntemleri.....	<b>7</b>
2.1.2. Endüstride anomali tespiti .....	<b>10</b>
2.2. Denetimli Öğrenme.....	<b>12</b>
2.3. Pekiştirmeli Öğrenme .....	<b>14</b>
2.3.1. Çok kollu slot .....	<b>16</b>
2.3.1.1. UCB algoritması .....	<b>18</b>
2.3.2. Çok kollu slot ve anomali tespiti .....	<b>19</b>
2.4. İstatiksel Proses Kontrol .....	<b>20</b>
2.4.1. İstatiksel proses kontrol çeşitleri .....	<b>21</b>
2.4.2. Kontrol grafikleri.....	<b>21</b>
2.4.3. Kontrol grafiklerinde örüntü tanıma.....	<b>26</b>
<b>3. MATERYAL VE YÖNTEM</b> .....	<b>35</b>
3.1. Materyal.....	<b>35</b>
3.1.1. Laboratuvar ortamı .....	<b>35</b>
3.1.2. Benzetim ortamı .....	<b>37</b>
3.2. Akıllı Fabrikalarda OTA'ların Oluşturduğu Trafikteki Anormal Durumların Tespiti.....	<b>39</b>
3.2.1. Karar ağacı ile anomali tespiti.....	<b>39</b>
3.2.2. Bulanık mantık tabanlı anomali tespiti.....	<b>41</b>
3.3. OTA'lar için Konumlandırma Sistemindeki Anormal Durum Tespiti.....	<b>44</b>
<b>4. BULGULAR VE TARTIŞMA</b> .....	<b>57</b>
4.1. Akıllı Fabrikalarda OTA'ların Oluşturduğu Trafikte Anormal Durum Tespiti Sonuçları .....	<b>57</b>
4.1.1. Karar ağacı ile anomali tespiti sonuçları .....	<b>57</b>
4.1.2. Bulanık mantık tabanlı anomali tespiti sonuçları .....	<b>59</b>
4.2. OTA'lar için Konumlandırma Sistemindeki Anormal Durum Tespiti Sonuçları .....	<b>62</b>
<b>5. SONUÇ VE ÖNERİLER</b> .....	<b>68</b>
<b>KAYNAKLAR DİZİNİ</b> .....	<b>69</b>



## ŞEKİLLER DİZİNİ

<u>Sekil</u>	<u>Sayfa</u>
2.1. Nokta anomali .....	5
2.2. Bağlamsal anomali .....	5
2.3. Toplu anomali.....	6
2.4. Anomali tespiti kullanım alanları .....	7
2.5. Anomali tespiti taksonomi.....	7
2.6. Anomali tespitinde makine öğrenmesi ve istatistiksel yaklaşımlar .....	8
2.7. ÇKS kullanım alanları .....	17
2.8. ÇKS taksonomisi.....	18
2.9. Kontrol grafiği çeşitleri .....	22
2.10. CCP çeşitleri: (a) normal, (b) katmanlaşma, (c) sistematik, (d) döngüsel, (e) yukarı kayma, (f) aşağı kayma, (g) artan trend, (h) azalan trend .....	26
3.1. (a) IFARLAB Akıllı fabrika ortamı ve OTA, (b) IFARLAB ortam ve yol şeması .....	36
3.2. GAZEBO ortamında oluşturulmuş akıllı fabrika sanal ortamı .....	37
3.3. Anomali tespiti akış diyagramı: (a) sistem eğitimi, (b) test verisi üzerinden anomali tespiti .....	41
3.4. Giriş değerleri ve çıkış değeri üyelik fonksiyonları.....	43
3.5. Pekiştirmeli öğrenme tabanlı veri odaklı önerilen anomali tespiti akış şeması .....	46
3.6. CCPR modelinin oluşturulması için akış şeması .....	53
4.1. C# Uygulaması simülasyon ortamında normal durum .....	58
4.2. C# Uygulaması simülasyon ortamında farklı yol parçalarında tespit edilen durma ve yaya yoğunluğu kaynaklı yavaşlama anomalileri .....	58
4.3. C# Uygulaması simülasyon ortamında farklı yol parçalarında tespit edilen araç kaynaklı bilinmeyen ve yavaşlama anomalileri.....	59
4.4. Pozitif gerçek durum “var” için ROC analiz grafiği.....	60
4.5. Pozitif gerçek durum “yok” için ROC analiz grafiği.....	61
4.6. Test 1 için UCB algoritmasında elde edilen grafikler, (a) <i>KOL3</i> için ortalama ödül grafiği, (b) <i>KOL2</i> için ortalama ödül grafiği, (c) <i>KOL1</i> için ortalama ödül grafiği, (d) <i>KOLO</i> için ortalama ödül grafiği, (e) genel ortalama ödül grafiği, (f) <i>KOLO</i> , <i>KOL1</i> , <i>KOL2</i> ve <i>KOL3</i> ’ün seçilme sayılarının normal ve gürültülü veriye göre değişimi (normal veri mavi renk, gürültülü veri kırmızı renk) .....	63
4.7. Test 2 için UCB algoritmasında elde edilen grafikler, (a) <i>KOL3</i> için ortalama ödül grafiği, (b) <i>KOL2</i> için ortalama ödül grafiği, (c) <i>KOL1</i> için ortalama ödül grafiği, (d) <i>KOLO</i> için ortalama ödül grafiği, (e) genel ortalama ödül grafiği, (f) <i>KOLO</i> , <i>KOL1</i> , <i>KOL2</i> ve <i>KOL3</i> ’ün seçilme sayılarının normal ve gürültülü veriye göre değişimi (normal veri mavi renk, gürültülü veri kırmızı renk) .....	64
4.8. Test 3 için UCB algoritmasında elde edilen grafikler, (a) <i>KOL3</i> için ortalama ödül grafiği, (b) <i>KOL2</i> için ortalama ödül grafiği, (c) <i>KOL1</i> için ortalama ödül grafiği, (d) <i>KOLO</i> için ortalama ödül grafiği, (e) genel ortalama ödül grafiği, (f) <i>KOLO</i> , <i>KOL1</i> , <i>KOL2</i> ve <i>KOL3</i> ’ün seçilme sayılarının normal ve gürültülü veriye göre değişimi (normal veri mavi renk, gürültülü veri kırmızı renk).....	64
4.9. Test 1 için önerilen sistem ile yapılan anomali tespiti .....	65
4.10. Test 2 için önerilen sistem ile yapılan anomali tespiti.....	66
4.11. Test 3 için önerilen sistem ile yapılan anomali tespiti.....	66

## ÇİZELGELER DİZİNİ

<b><u>Cizelge</u></b>	<b><u>Sayfa</u></b>
2.1. Değişken kontrol grafikleri kontrol limiti hesaplanma formülleri ve açıklamaları.....	24
2.2. CCP Çeşitlerinin üretiminde kullanılacak formüller.....	27
2.3. CCP tanıma için kullanılan bazı özniteliklerin adı, formül açıklaması ve formülü .....	30
3.1. Veri kümesine ait öznitelikler .....	38
3.2. Veri kümesinin kullanıldığı yöntem ve teknolojiler.....	39
4.1. Yol parçasının içeriği ve test doğruluk oranları .....	59
4.2. Bulanık mantık tabanlı anomali tespiti değerlendirme metrikleri.....	61

## SİMGELER VE KISALTMALAR DİZİNİ

<b><u>Kısaltmalar</u></b>	<b><u>Açıklama</u></b>
IFARLAB	Akıllı Fabrika ve Robotik Laboratuvarı (Intelligent Factory and Robotic Laboratory)
OTA	Otonom Taşıyıcı Araç
CCP	Kontrol Grafiği Örüntüsü (Control Chart Pattern)
CCPR	Kontrol Grafiği Örüntü Tanıma (Control Chart Pattern Recognition)
UCB	Üst Güven Sınırı (Upper Confidence Bound)
DKG	Dinamik Kontrol Grafiği
ÇKS	Çok Kollu Slot
İPK	İstatiksel Proses Kontrol
ÜKL	Üst Kontrol Limiti
AKL	Alt Kontrol Limiti
ARL	Ortalama İşlem Uzunluğu (Average Run Length)
AMCL	Uyarlanabilir Monte Carlo Yerelleştirme (Adaptive Monte Carlo Localization)
UWB	Ultra Geniş Bant (Ultra-wideband)
MM	Mahalanobis Mesafesi (Mahalanobis Distance)
HM	Hausdorff Mesafesi (Hausdorff Distance)
LS	En Küçük kare (Least squares)
SS	Standart Sapma
MSE	Hata Kareleri Ortalama Toplamı (Mean Sum of Squares of Errors)
ROC	Alıcı Operatör Karakteristikleri Eğrisi (Receiver Operator Characteristics Curve)

## 1. GİRİŞ VE AMAÇ

Teknolojinin ilerlemesi ve artan dijitalleşme ile birlikte endüstri alanında da yenilikler ve gelişmeler meydana gelmektedir. Bu gelişmeler Endüstri 4.0 kavramı ile bütüncül bir bakış açısı altında hız kazanmıştır (Ghobakhloo, 2020). Bu yaklaşımda insan için zorlayıcı ve rutin olan işlerin otonom sistemler tarafından yapılması hedeflenmektedir (Osterrieder vd., 2020). Bu amaçla akıllı fabrikalar için gerekli sistemlerin oluşturulması ve kullanımı konusunda çalışmalar yoğunlaşmaktadır. Çalışma alanlarından biri de sistem sürdürülebilirliği ve kontrolü için veri odaklı anomali tespiti çalışmalarıdır.

Endüstride veri odaklı anomali tespiti ile ilgili çalışmalar bulunmaktadır (Khalastchi vd., 2015; Stojanovic vd., 2016; Stojanovic vd., 2017; Wang vd., 2017; Pan, 2017; Liang vd., 2019). Ancak endüstri 4.0, akıllı fabrikalarda otonom taşıyıcı araç (OTA)'lar için anomali tespiti ile ilgili literatürde çok fazla çalışma bulunmamakla birlikte gelişmekte olan bir alandır (Örnek vd., 2018; Örnek vd., 2020; Değirmenci vd., 2020). Örnek vd. (2018), denetimli öğrenme kullanarak ortam veya OTA kaynaklı oluşabilecek anormal durumları belirleyerek akıllı fabrikalardaki OTA için anomali tespiti gerçekleştirmişlerdir. Örnek vd. (2020), OTA için bulanık mantık tabanlı anomali tespiti ile anomali “var” veya “yok” olarak durum tespiti gerçekleştirmişlerdir. Değirmenci vd. (2020), pekiştirmeli öğrenme ile akıllı fabrikalarda OTA'ların kullandığı kavşakların karakteristiğini öğrenerek, karakteristiğin değişmesi durumunu anormal durum olarak tespit etmişlerdir.

Literatürde, endüstri için veri odaklı anomali tespiti kapsamında denetimli, denetimsiz, yarı-denetimli ve pekiştirmeli öğrenme tabanlı yöntemler kullanılabilir. Denetimli öğrenme ile anomali tespiti gerçekleştirilebilmesi için veri kümesinin etiketli olması yani anormal ve normal durumların bilinmesi gerekmektedir. Ancak etiketli veri kümesi oluşturmak bazı durumlar için maliyetli, zaman alıcıdır. Ayrıca oluşturulan etiketli veri kümesi kimi zaman anormal durumların tespit edilmesinde, yeni anormallikleri bilemediği için, yanlış sonuçlar verebilmektedir. Bu gibi nedenlerle normal veya anormal durumların bilinmediği denetimsiz öğrenme ve sadece normal durumların bilindiği yarı-denetimli öğrenme ile anomali tespiti yöntemleri de kullanılmaktadır. Pekiştirmeli

öğrenme altında yer alan Çok Kollu Slot (ÇKS) yöntemleri ise karakteristiğın öğrenilmesi ve karakteristiğe aykırı durumların anomali olarak tespit edilmesinde kullanılmıştır (Değirmenci vd., 2020).

Fabrikalarda OTA'lar tarafından gerçekleştirilen taşıma işlemlerinde oluşabilecek bir problem üretim sisteminin durmasına neden olabilir veya sistemin verimliliğini etkileyebilir. Bu nedenle, bu tez çalışmasında OTA'lar için fabrika seviyesi oluşan trafikteki anormal durum ve OTA seviyesinde ise konumlandırma sistemlerinde oluşan anormal durum tespiti olmak üzere iki farklı problem ele alınarak veri üzerinden anormal durum tespiti gerçekleştirilmiştir. İlk problemde fabrika seviyesinde OTA'ların oluşturduğu trafiğın izlenerek meydana gelebilecek anormal durumların tespiti yapılmıştır. Burada, denetimli öğrenme yöntemlerinden karar ağacı ve bulanık mantık tabanlı yöntem kullanılmıştır. Gazebo benzetim ortamında oluşturulan akıllı fabrika ortamı baz alınarak OTA'ların kullandığı fabrika içerisindeki yollarda meydana gelebilecek anormal durumların tespiti yapılmıştır. Karar ağacı kullanılarak; yaya geçitlerinde oluşan beklenmeyen yoğunluk, doldurma boşaltma noktalarında beklenmeyen durma, OTA kaynaklı yavaşlama veya OTA kaynaklı nedeni tam bilinmeyen anormal durumların tespiti OTA ve ortam bilgileri kullanılarak gerçekleştirilmektedir. Bulanık mantık yönteminde ise OTA ve ortam bilgileri kullanılarak anomali “var” veya “yok” olarak tespit gerçekleştirilir. İkinci problemde ise OTA'ların davranışlarını otonom olarak gerçekleştirmeleri için kritik öneme sahip konumlandırma sistemindeki anormal durum tespitine odaklanılmıştır. OTA'ların fabrika içi farklı yollardaki hareketi sırasında kullandığı Ultra Geniş Bant (Ultra-wideband, UWB), Uyarlanabilir Monte Carlo Yerelleştirme (Adaptive Monte Carlo Localization, AMCL) ve Odometre olmak üzere üç farklı konumlandırma sisteminden gelen konum bilgisinde zaman içinde olan anormal değişimin tespiti yapılmaktadır. Burada öncelikle, verilen bir yol parçası için konumlandırma sistemlerine ait karakteristikler pekiştirmeli öğrenme altında bulunan ÇKS yaklaşımında yer alan Üst Güven Sınırı (Upper Confidence Bounds, UCB) algoritması ile öğrenilmektedir. Öğrenmedeki ödül değişimleri İstatistiksel Proses Kontrol (İPK)'deki farklı yaklaşımlar ile takip edilerek karakteristik değişimlerinin anomali veya normal olma durumu tespit edilmektedir. Önerilen çalışma Akıllı Fabrika ve Robotik Laboratuvarı (Intelligent Factory and Robotic Laboratory, IFARLAB)'nda yer alan OTA'dan alınan veriler kullanılarak test edilmiştir.

Bölüm 2’de anomali tespiti, denetimli öğrenme, pekiştirmeli öğrenme ve İPK ile ilgili temel literatür verilmektedir. Bölüm 3’de kullanılan materyal bilgileri ve önerilen anormal durum tespit yaklaşımları verilmektedir. Bölüm 4’te ise fabrika seviyesindeki OTA trafiğinde ve OTA seviyesinde konumlandırma sistemlerindeki anomali tespiti sonuçları verilmektedir. Son olarak bölüm 5’te sonuç ve öneriler verilmektedir.

## 2. LİTERATÜR ARAŞTIRMASI

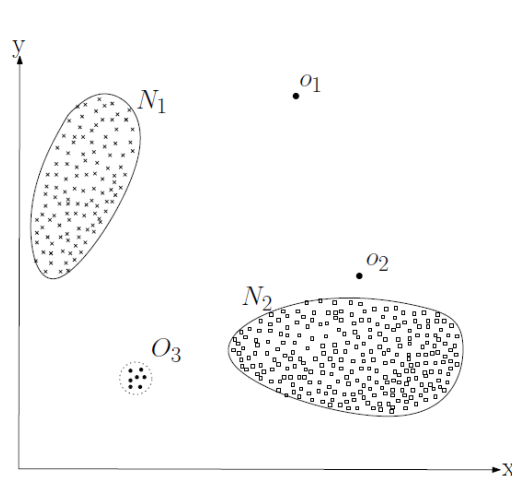
Bu bölümde ilk olarak anomali tespiti altında anomali nedir, anomali tipleri, anomali tespit yöntemleri ve endüstride anomali tespiti çalışmaları verilmektedir. Devamında denetimli öğrenme, pekiştirmeli öğrenme, pekiştirmeli öğrenme altında ÇKS, UCB algoritması ve ÇKS ile anomali tespiti çalışmaları verilmektedir. Son olarak İPK, İPK çeşitleri, kontrol grafikleri ve kontrol grafiği örüntü tanıma (Control Chart Pattern Recognition, CCPR) ile ilgili temel literatür verilmektedir.

### 2.1. Anomali Tespiti

Anomali, veri kümesindeki örüntüye uymayan yapılara denmektedir. Her bir veri kümesi kendi içerisinde belli bir dağılım, desen, örüntü içermektedir. Veri kümesinde daha sonradan oluşabilecek veya içerisinde zaten bulunmakta olan çoğunluğa uymayan veriler bulunabilmektedir. Bu veriler anormal, aykırı veya anomali olarak tanımlanmaktadır. Anomaliler tiplerine göre üçe ayrılmaktadır (Chandola vd., 2009):

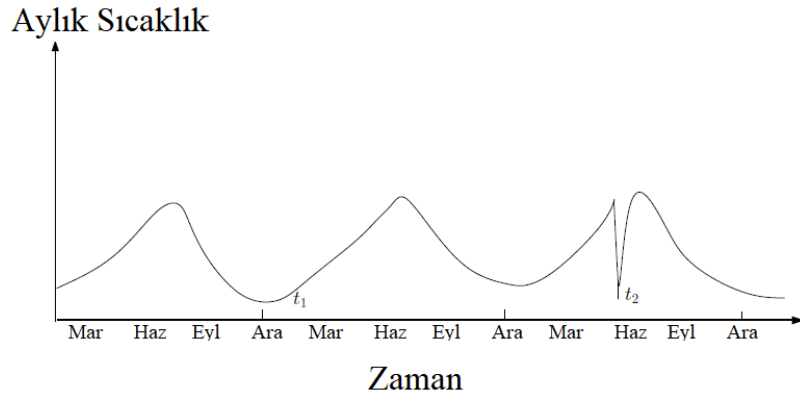
- Nokta Anomali
- Bağlamsal Anomali
- Toplu Anomali.

Nokta anomali, bir veri kümesinde tek bir veri örneğinin anormal olma durumudur. Şekil 2.1’de nokta anomaliye örnek verilmektedir. Şekilde gösterilen  $N_1$  ve  $N_2$  kümelerine dahil noktalar belirli bir büyüklük ve çoğunluğa sahipken  $o_1$ ,  $o_2$  noktaları ve  $O_3$  kümesindeki noktalar çoğunluktan ayrı ve azınlık olarak bulunmaktadır. Bu nedenle  $N_1$  ve  $N_2$  kümeleri normal,  $o_1$ ,  $o_2$  noktaları ve  $O_3$  kümesindeki noktalar nokta anomali olarak kabul edilmiştir.



Şekil 2.1. Nokta anomali (Chandola vd.'den, 2009)

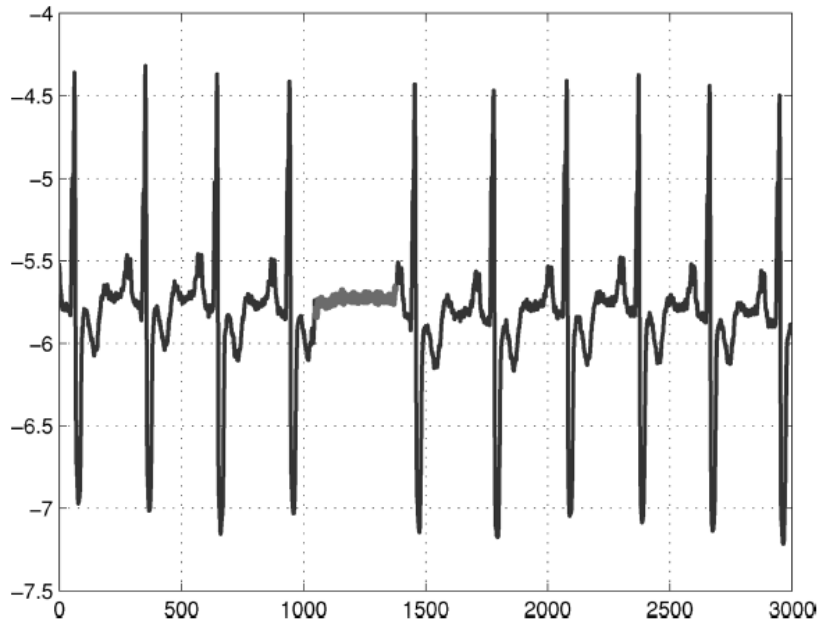
Bağlamsal anomali, veri kümesi içerisinde belli bir bağlama göre verinin anormal olmasıdır. Bu durumu şu şekilde örnekleyebiliriz; her yıl aylara ait hava sıcaklıklarının tutulduğu bir veri kümesinde yaz aylarına ait sıcaklık değerlerinin normalde yüksek olduğunu kabul edelim. Eğer yaz aylarına ait sıcaklık değerinde beklenmeyen bir düşme meydana gelirse bu durum yaz ayları sıcaklık değerlerine göre anormaldir. Ancak aynı değer yaz ayları soğuk geçen bir yerde meydana gelseydi anormal olarak kabul edilmeyecekti. Şekil 2.2’de bağlamsal anomali örneğine ait grafik verilmektedir.



Şekil 2.2. Bağlamsal anomali (Chandola vd.'den, 2009)

Toplu anomali ise veri kümesinde bir grup verinin bir araya gelmesi sonucunda oluşan anomalidir. Örnek olarak kalp ritmi çizelgesi verilebilir. Kalp ritmi sinyallerinde belirli bir veri bloğunun oluşması sağlıklı bir kişi için anormal durumdur. Şekil 2.3’te bu anomaliye ait örnek verilmektedir.



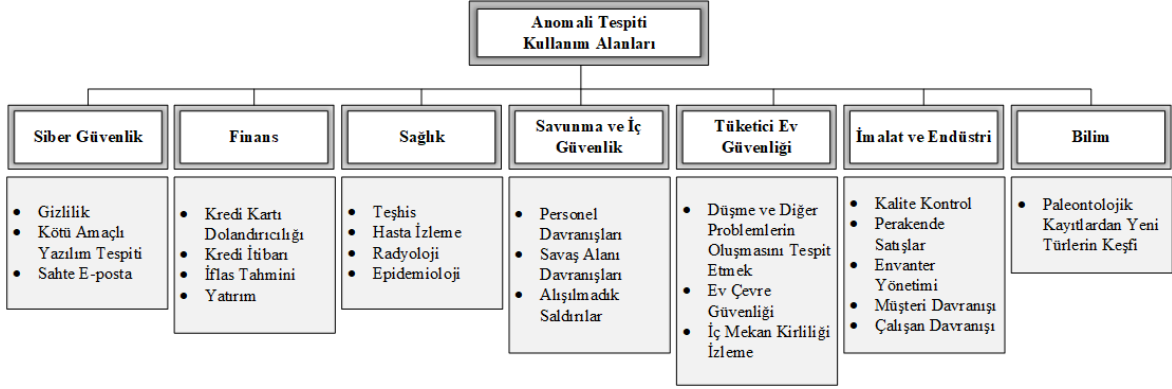


Şekil 2.3. Toplu anomali (Chandola vd.'den, 2009)

Anomaliler istenmeyen durumlar olarak açıklanır. Bu istenmeyen durumlar için iki durum söz konusudur. Bunlar:

- Veri kümesindeki örüntüyü bozan anormal verilerin ana veriyi iyileştirme amacıyla gürültü olarak kabul edilip atılması veya,
- Anormal verilerin öğrenilmesi ile bir daha oluşma durumunda erken tespit edilerek sistemin korunmasının sağlanmasıdır.

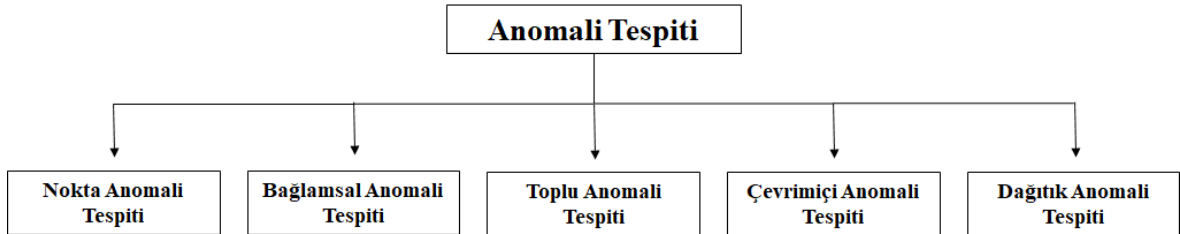
Sistemlerin çalışmaları sırasında ürettikleri veriler kullanılarak sistemde oluşabilecek anormal durumların, anomalilerin tespit edilmesi ise “anomali tespiti” olarak tanımlanmaktadır. Anomali tespiti birçok farklı alanda farklı amaçlar için kullanılmaktadır. Literatürde anomali tespiti kapsamında siber güvenlik (Evangelou ve Adams, 2020), finans (Vanhoeyveld vd., 2020), sağlık (Sarmadi ve Karamodin, 2020), savunma ve iç güvenlik (Macdonald vd., 2020), imalat ve endüstri (Huang vd., 2020) alanlarında çalışmalar bulunmaktadır. Şekil 2.4’te anomali tespitinin kullanıldığı alanlardan bazı örnekler verilmektedir (Mehrotra vd., 2017).



Şekil 2.4. Anomali tespiti kullanım alanları (Mehrotra vd., 2017)

### 2.1.1. Anomali tespiti yöntemleri

Anomali tespitinin uygulandığı birçok farklı alan olduğu gibi gerçekleştirilmesi için kullanılan farklı yöntem ve algoritmalar bulunmaktadır. Banerjee vd. (2018), anomali tespiti taksonomisini anomali tiplerine ve kullanım durumuna göre tanımlamaktadır. Şekil 2.5'te taksonomi verilmektedir.



Şekil 2.5. Anomali tespiti taksonomi (Banerjee vd., 2018)

Bağlamsal anomali tespitinde, veri kümesindeki verilerin birbirleriyle ilişkisi ön plandadır. Bu nedenle verilerin birbiriyle olan ilişkisini tanımlamak için kullanılan yöntemler, bu ilişkiye uymayan veriyi, bağlamsal anomaliyi, tespit etmek için kullanılabilir. Verinin zamana göre (çalışma günleri vb.) değerlendirilmesi anomaliyi ortaya çıkarabilir (Hayes ve Capretz, 2014).

Toplu anomali tespitinde, tek başına normal değerlendirilen ancak toplu halde bulunmaları durumunda anormal durum olabilecek durumların tespit edilmesidir. Toplu anomali tespitinde farklı durumların tespiti için farklı yöntemler bulunmaktadır:

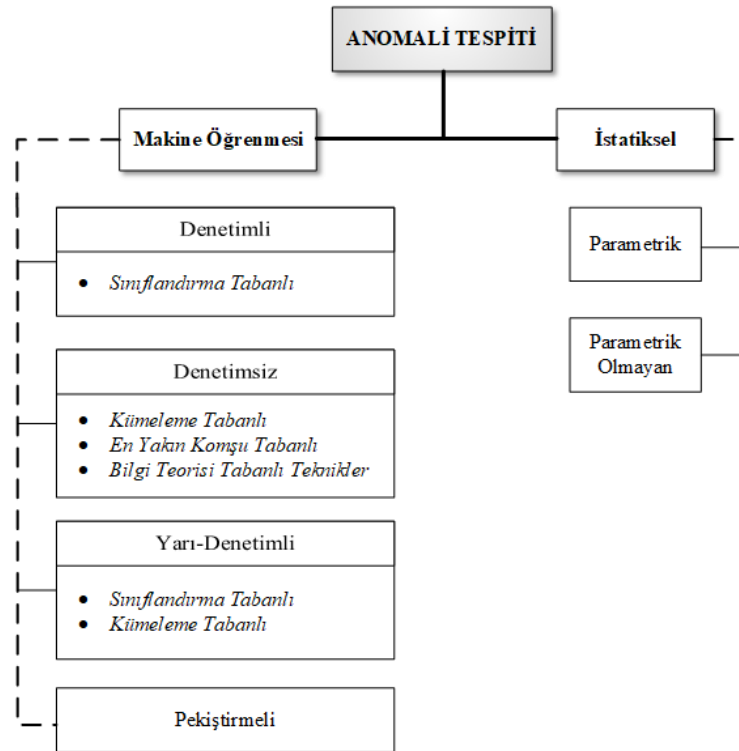
- Anormal sıralı durumların tespiti: ardışık anomali tespiti,

- Mekansal veri kümesinde anormal olan alt alanların tespiti: mekânsal anomali tespiti ve
- Çizge verilerindeki anormal alt-çizgelerin tespit edilmesi: çizge anomalisi tespitidir (Chandola vd., 2009).

Çevrimiçi anomali tespiti, veri akışının devam ettiği gerçek zamanlı durumlarda anormal durumların tespit edilmesi için kullanılır. Çevrimiçi anomali tespiti yöntemleri, verinin temel özelliklerinde oluşabilecek durumlara zaman içinde uyum sağlayabilir (Chandola vd., 2009).

Dağıtık anomali tespiti, anormal durumların birden fazla konumda oluşması ve tek bir noktadan anormal durumların tespitinin zor olduğu durumlarda kullanılabilir (Chandola vd., 2009).

Literatürde anomali tespiti için amacına, kullanım alanına, anomali tipine bağlı farklı taksonomiler bulunabilir. Şekil 2.6’da verildiği üzere makine öğrenmesi ve istatistiksel yaklaşımlar olmak üzere iki grupta da verilebilir.



Şekil 2.6. Anomali tespitinde makine öğrenmesi ve istatistiksel yaklaşımlar

Makine öğrenmesi yöntemleri; denetimli öğrenme, yarı denetimli öğrenme, denetimsiz öğrenme ve pekiştirmeli öğrenme yöntemleridir. Denetimli öğrenme, veri kümesinde verilerin etiketli olması yani anormal ve normal durumların bilinmesi durumunda kullanılmaktadır. Sınıflandırma tabanlı yöntemler kullanılmaktadır. Sınıflandırma tabanlı yöntemlerde etiketli veri kümesi kullanılarak model oluşturulur. Oluşturulan model kullanılarak test gerçekleştirilir. Test sonucunda veri öğrenilen sınıflardan birine atanır. Sınıflandırma tabanlı yöntemler ikiye ayrılabilir (Chandola vd., 2009):

- Model tabanlı: çok sınıflı anormallik saptamalarda:
  - Sinir ağları tabanlı
  - Bayes ağları tabanlı
  - Destek vektör makineleri tabanlı yöntemler kullanılmaktadır.
- Kural tabanlı: Normal davranışını kurallar ile tanımlar, kuralara uymayan durumları anomali olarak tespit eder. Kural tabanlı yaklaşımlar ise Bulanık mantık ve Genetik algoritmalar gibi yöntemlerdir.

Denetimsiz öğrenme, veri kümesindeki verilerin etiketsiz olarak bulunması durumunda verinin normal veya anormal olduğunun bilinmediğini kullanılmaktadır. Denetimsiz öğrenmede kullanılan yaklaşımlar (Chandola vd., 2009):

- Kümeleme: etiketlenmemiş bir veri kümesinde kullanılır. Veri kümesindeki normal verilerin birlikte sayıca fazla olarak kümelenmesi, anormal verilerin ise küçük sayıda kümelenmesi veya kümeye dahil olmaması baz alınarak verilerin kümelenme biçimlerine göre anormal durumlar tespit edilir. İkiye ayrılabilir:
  - En yakın komşu: her bir veri için komşu değerleri hesaplanır, daha sonra bir verinin anormal olup olmadığını anlamak için komşu değerlerini analiz eder. İkiye ayrılabilir:
    - Mesafeye dayalı yöntemler: normal noktalara yakın komşu noktalar varken anormal noktalar komşu noktalardan uzaktadır.
    - Yoğunluk temelli yöntemler: anormal veriler düşük yoğunluklu alanlarda bulunur.
  - Bilgi teorisi tabanlı teknikler (Chandola vd., 2009).

Yarı denetimli öğrenme, veri kümesinde sadece normal verilerin bilinmesi durumunda kullanılmaktadır. Sınıflandırma ve kümeleme tabanlı teknikler kullanılmaktadır.

Pekiştirmeli Öğrenme, değişken bir ortamda ortam karakteristiğinin deneme yanılma yöntemiyle aldığı kararlar karşılığında ödül veya ceza olarak öğrenim gerçekleştirmesi üzerinedir. Öğrenilen karakteristiğe aykırı oluşacak karakteristikler veya durumlar anomali olarak değerlendirilebilir. Değişken ortamlarda anomali tespiti için etiketli veri oluşturmanın maliyeti, zorluğu ve veri kümesi güncelliğinin korunması nedeniyle denetimli öğrenmeye göre avantajlıdır.

İstatiksel yaklaşımlar parametrik ve parametrik olmayan olarak ikiye ayrılır. Parametrik olmayan teknikler dağılımın öğrenilmesi için kullanılırken parametrik tekniklerde ise normal verilerin bir dağılıma sahip olduğu kabul edilir. Parametrik tekniklerde bu dağılım kullanılarak anomaliler tespit edilir.

### **2.1.2. Endüstride anomali tespiti**

Endüstriyel sistemlerde sistemlerin kesinti içermeden çalışması verimlilik ve zaman yönetimi açısından önemlidir. Bu nedenle endüstriyel sistemlerde kesintilere sebep olabilecek anormal durumların, insan veya sistem kaynaklı, tespit edilmesi hem güvenlik hem de üretim açısından gereklidir. Literatürde endüstriyel sistemlerde; üretim vb. anormal durum tespit çalışmaları bulunmaktadır (Freeman vd., 2013; Khalastchi vd., 2015; Stojanovic vd., 2016; Pan, 2017; Stojanovic vd., 2017; Wang vd., 2017; Banjanovic-Mehmedovic vd., 2017; Liang vd., 2019; Fahim ve Sillitti, 2019).

Freeman vd. (2013), insansız hava aracı için model tabanlı ve veri odaklı hata tespitini ele almışlardır. Model tabanlı hata algılama, rüzgar gibi dışsal bozuklukları reddetmek ve model hatalarına karşı sağlamlık sağlamak için dayanıklı doğrusal filtreleme yöntemlerini kullanır. Veriye dayalı algoritma, ayrıntılı sistem bilgisi olmadan yalnızca ham uçuş testi verileri üzerinde çalışmak üzere geliştirilmiştir. Khalastchi vd. (2015), otonom robotlar için mahalnobis uzaklığı ile çevrimiçi veri odaklı anormallik algılama yaklaşımı sunmuşlardır. Yaklaşım, yalnızca çevrimiçi olarak toplanan verilere dayanarak

bir hata bildirdiği için otonom robotların dinamik doğasına uygun ve denetimsiz, modelden bağımsız ve alandan bağımsızdır. En son  $m$  adet giriş örneklerine bakılarak (kayan pencere) bu çevrimiçi verilerin bir arıza içerip içermediğine karar verilir. Üç adımı bulunmaktadır: giriş verileri gürültüyü azaltmak için filtrelenir, verileri ilişkili öznitelik kümelerine bölerek boyut küçültme uygulanır ve her grup için aykırı değer olma derecesini hesaplamak için mahalnobis uzaklığı kullanılır. Hesaplanan değer belirlenen eşik üzerinde ise bu durum anormallik olarak tespit edilir. Yaklaşımlarını, ticari insansız hava araçları, vakumlu temizleme robotu, yüksek kaliteli bir uçuş simülatörü ve bir uzay aracının elektrik güç sisteminde test etmişlerdir. Stojanovic vd. (2016), çalışan sistemde ortaya çıkabilecek anormalliklerin çok boyutlu analizini ve gerçek zamanlı tespitini sağlayan endüstri süreçlerinde veri odaklı kalite yönetimini önermişlerdir. İki ana adımdan oluşur: geçmiş verilere dayanarak sistemin normal davranışını öğrenmek ve gerçek zamanlı verileri işleyerek anormal bir davranışı gerçek zamanlı olarak tespit etmek. Sistemin normal davranışı, “Hadoop” gibi toplu işleme için büyük veri teknolojileri kullanılarak geçmiş verilerden hesaplanır. Sistemin olağan davranışından sapmaları ve eğilimlerini hesaplamak için ise “Storm” gibi gerçek zamanlı işleme teknolojilerini kullanmışlardır. Wang vd. (2017), jeneratör ayar noktalarının kötü niyetli manipülasyonlarını tespit etmek için veriye dayalı bir çevrimiçi mekansal anormallik tespiti metodolojisini önermişlerdir. Temel olarak Otomatik Üretim Kontrolü'nü hedefleyen veri bütünlüğü saldırısını ve siber izinsiz girişler yoluyla jeneratör ayar noktalarına kötü niyetli manipülasyonu dikkate almışlardır. Yaklaşımın bir parçası olarak, büyük hacimli ham verilerden en kritik bilgileri çıkarmak için jeneratör davranış uygunluk ölçütlerini tanımlamışlardır. Her bir davranış modeli örneğini çevrimdışı gruplamak için yarı denetimli  $K$ -ortalımalı kümelemeyi kullanmışlardır. Çevrimdışı elde edilen modelin çevrimiçi uygulama sırasında döngüsel olarak sınıflandırma için kullanılması ile anormallik algılama gerçekleştirilmiştir. Banjanovic-Mehmedovic vd. (2017), termik santralin seçilen buhar kızdırıcıları ve buhar kazanı bölümlerinde anormallik tespiti için kullanılacak sinir ağı tabanlı veri odaklı bir model üzerine çalışmışlardır. Sinir ağları için girdiler, bu bölümlerin en önemli işlem değişkenlerinden bazılarıdır. Buhar Kızdırıcıları için; aşırı ısıtılmış buhar sıcaklıkları, aşırı ısıtılmış buhar akışları, kızgın buhar soğutma suyu akışları kullanılmıştır. Buhar kazanı için ise; kazan seviyeleri, kazan basınçları, besleme suyu akışları kullanılmıştır. Tüm girdiler termik santralin izleme sisteminden gözlemlenmiş ve anormal / normal davranışları operatörün deneyimleriyle tanımlanmıştır.

Anomali algılama sorununu çözmek için üç farklı türde sinir ağı (MLP, tekrarlayan ve olasılıklı) kullanışlardır. Liang vd. (2019), bilgisayarlı sayısal kontrol işleme süreçlerindeki anormallikleri verimli bir şekilde tespit etmek için veriye dayalı tanı sistemi sunmuşlardır. İşleme süreçleri sırasında çevrimiçi olarak toplanan güç verileri ön işleme tabii tutulur. Ön işlemede, izlenen verilerin gürültüsünün giderilmesi, normalleştirme ve hizalama işlemlerini gerçekleştirmişlerdir. Daha sonra işlenmiş verileri kullanarak anormallikleri tanımlamak için eşikler tanımlamışlardır. Makineyle işleme proseslerinin dinamik koşulları nedeniyle anormallikleri tanımlamak için kullanılan eşiklerin değişebileceği belirtmişlerdir. Bu dinamik koşullarda daha doğru algılama elde etmek için optimum anormallik eşiklerini belirlemek için bir meyve sineği optimizasyonu (fruit fly optimization) algoritması kullanmışlardır. Fahim ve Sillitti (2019), endüstriyel sistemlerde kullanılan anomali tespiti yöntemleri ve çalışmaları ile ilgili inceleme sunmaktadır. Kullanılan yöntem, veri türü, tespit edilen anomali tipi, verilerin kullanılabilirliği, sensör tipleri, uygulama alanları ve değerlendirme kriterleri ile ilgili bilgiler vermişlerdir. Nguyen vd. (2017), yakıt hücreli araç için hava kompresörü motor hız verilerini zamansal mantık (temporal logic), Zhao vd. (2017), fabrikalardaki elektrik jeneratörler için jeneratörde bulunan kademe sensörü verilerini korelasyon analizi, Akiyama vd. (2015), güneş enerjisi üretim sistemlerinde elektrik akım verisini yoğunluk fonksiyonu modeli (density function model), Zang vd. (2018), petrol boru hattında basınç sensör verilerini Markov zinciri, Yan (2016), elektrik santralinde sıcaklık sensör verilerini aşırı öğrenme makineleri (extreme learning machines) ve Wu ve Chen (2018), rüzgar türbinleri için 150 rüzgar türbini ölçüm parametresini Koşullu Gradyan artırıcı karar ağacı (Conditional Gradient boosting decision tree) yönteminde kullanarak anomali tespiti gerçekleştirmişlerdir (Fahim ve Sillitti, 2019).

## 2.2. Denetimli Öğrenme

Denetimli öğrenme, girdi öznitelikleri ve bir hedef öznitelik arasındaki ilişkiyi keşfetmeye çalışan yöntemlerdir (Maimon ve Rokach, 2005). Bu ilişki, model olarak adlandırılır. Model, veri setindeki giriş özniteliklerini kullanarak hedef özniteliğin değerini tahmin etmek için kullanılabilir olacak olayları tanımlamaktadır. Modelin elde edilmesi için gerekli en önemli bileşen veri kümesidir. Denetimli öğrenme yöntemlerinde etiketli veri

kümeleri kullanılmaktadır. Etiketli veri kümelerinde hedef özniteliği bilinmektedir. Hedef öznitelikler “sınıf” olarak da adlandırılmaktadır.

Denetimli yöntemler ikiye ayrılabilir. Bunlar; sınıflandırma ve regresyon modelleridir. Sınıflandırma, veri kümesindeki sınıf özniteliğinin kesikli değerler içermesi durumunda kullanılmaktadır. Sınıflandırma altında:

- Destek vektör makineleri,
- Naive bayes,
- En yakın komşu ve
- Sinir ağları
- Karar ağaçları ve
- Lojistik regresyon yöntemleri verilebilir.

Sınıflandırma sınıf özniteliğinin içeriğine göre ikiye ayrılır. Bunlar; ikili ve çoklu sınıftır. İkili sınıf, sınıf özniteliğinin içeriğinde sadece iki farklı durum olduğunu belirtir. Örnek olarak “var” ve “yok” verilebilir. Çoklu sınıf ise sınıf özniteliğinde ikiden fazla farklı durum olduğunu belirtmektedir. Diğer denetimli yöntemlerden regresyon ise veri kümesindeki sınıf özniteliğinin kesiksiz değerler içermesi durumunda kullanılmaktadır. Regresyon altında:

- Lineer regresyon,
- Hiyerarşik,
- Karar ağaçları ve
- Sinir ağları yöntemleri verilebilir.

Denetimli öğrenme iki aşamadan oluşmaktadır. Bu aşamalar eğitim ve test olarak adlandırılmaktadır. Bu aşamalarda etiketli veri kümesinden elde edilen eğitim ve test veri kümeleri kullanılmaktadır. İlk aşama eğitimde etiketli eğitim veri kümesi denetimli öğrenme yöntemleri ile kullanılarak sınıf özniteliğinin tespiti için model elde edilir. İkinci aşama olan testte ise elde edilen model etiketli test veri kümesinde giriş özniteliklerinde kullanılarak sınıf özniteliğinin tahmini gerçekleştirilir. Daha sonra elde edilen modelin test edilmesi sonucunda ne kadar başarılı olduğunu anlamak için çeşitli değerlendirme kriterleri kullanılmaktadır. Sınıflandırma yaklaşımlarının performans değerlendirilmesinde hata matrisi (Confusion Matrix) değerlendirme kriterleri kullanılabilir. Hata matrisinde: doğru-



pozitif, doğru-negatif, yanlış-pozitif ve yanlış-negatif değerleri hesaplanmaktadır. Sınıf değerleri “normal” veya “anormal” olan veri kümesi için açıklamalar şu şekilde verilebilir:

- Doğru-pozitif, test veri kümesindeki sınıf değeri “anormal” olan ve tahmin edilen sınıfı da “anormal” olan veri sayısını belirtir.
- Doğru-negatif, test veri kümesindeki sınıf değeri “normal” olan ve tahmin edilen sınıfı da “normal” olan veri sayısını belirtir.
- Yanlış-pozitif, test veri kümesindeki sınıf değeri “normal” olan ancak tahmin edilen sınıfı “anormal” olan veri sayısını belirtir.
- Yanlış-negatif: test veri kümesindeki sınıf değeri “anormal” olan ancak tahmin edilen sınıfı “normal” olan veri sayısını belirtir.

Regresyon kullanılması durumunda ise ortalama karekök sapması kullanılabilir. Ortalama karekök sapma değeri, test veri kümesindeki giriş değerleri için elde edilen tahmini kesiksiz sınıf değerleri ile gerçek kesiksiz sınıf değerleri farkı baz alınarak hesaplanmaktadır.

Denetimli yöntemler, pazarlama, finans ve üretim gibi çeşitli alanlarda uygulanabilir (Maimon ve Rokach, 2005). Bu alanlarda farklı amaçlar için kullanılmaktadır. Bu amaçlardan biri de anomali tespittir.

### 2.3. Pekiştirmeli Öğrenme

Pekiştirmeli öğrenme sistemlerin mevcut veri karakteristiğini kendi içinde öğrendiği ve etiketlemeye ihtiyaç duymadığı için farklı avantajlar sağlamaktadır (Kaelbling vd., 1996). Belirlenen hedeflere ulaşmak için aksiyonlar gerçekleştirilir. Bu aksiyonlar sonucunda çevre ile etkileşim kurulur ve sonucunda ortaya bir sonuç çıkar. Elde edilen sonucun hedefe ulaşılmasında sağladığı etki dikkate alınarak öğrenme gerçekleşir. Öğrenmeyi gerçekleştirirken çevre ile etkileşimde bulunulur. Örnek olarak küçük bir çocuğun bisiklete binmeyi öğrenmesini verebiliriz. Küçük çocuğun amacı bisikleti ilerletmektir. Yaptığı her aksiyon / hareket, ile çevre ile etkileşimde bulunur. Etkileşim sonucunda, yaptığı etki ile bir sonuç, tepki elde eder. Elde ettiği tepkiyi değerlendirerek doğru veya yanlış bir karar alıp almadığını öğrenir ve bir daha ki hareketini öğrendiklerini kullanarak alır.

Etkileşimden öğrenme, çoğu öğrenme ve zeka teorilerinde bulunan temel bir fikirdir (Sutton ve Barto, 2018). Bunlardan biri de makine öğrenmesi paradigmalarından biri olan pekiştirmeli öğrenmedir. Pekiştirmeli öğrenme, en fazla ödülü (doğru karar) elde etmek için durumların aksiyonlarla nasıl eşleştirileceğini öğrenmektir (Sutton ve Barto, 2018). Hangi hareketin seçilmesi gerektiği alınan ödüllere göre keşfedilir. Pekiştirmeli öğrenmede bulunan zorluklardan biri, keşif ve sömürü arasındaki dengeyi sağlamaktır. Keşif ve sömürü, öğrenme sırasında geçmişte ödül aldığı bilinen bir durumun seçilmesi mi yoksa gelecek için daha iyi olacak yeni bir durumun seçilmesi mi gerektiği olarak açıklanabilir. Bu dengeyi sağlamaya çalışmak için oluşturulmuş birçok farklı yöntem bulunmaktadır (Sutton ve Barto, 2018).

Pekiştirmeli öğrenme diğer makine öğrenmesi yöntemlerinden olan denetimli ve denetimsiz öğrenmeden farklıdır. Denetimli öğrenmede, etiketlenmiş veriler üzerinden öğrenme gerçekleştirilirken pekiştirmeli öğrenmede öğrenme kendi deneyimlerinden gerçekleştirilir. Denetimsiz öğrenme ise genellikle etiketlenmemiş veri içerisinden gizlenmiş bir örüntü bulma üzerineyken pekiştirmeli öğrenme ise gizlenmiş örüntüyü bulmaya çalışmak yerine en çok ödülü almaya çalışmaktadır.

Pekiştirmeli öğrenmenin altı adet bileşeni bulunmaktadır. Bunlar:

- Ajan
- Çevre
- Politika
- Ödül Sinyali
- Değer Fonksiyonu ve
- Çevre Modelidir

Ajan, pekiştirmeli öğrenmeyi gerçekleştiren algoritmadır. Çevre, öğrenme sırasında etkileşim halinde olunan, bulunulan ortamı temsil etmektedir. Politika, ajanın belirli bir zamanda çevre içerisindeki davranışını tanımlamaktadır. Ödül sinyali, ajanın çevrede gerçekleştirdiği her bir adım sonucunda değer almasıdır. Ajanın amacı öğrenme süresinde aldığı ödülleri en üst düzeye çıkartmaktır. Ödül sinyali anlık her adımda bir değeri ifade ederken değer fonksiyonu ise öğrenme sürecince ödül durumu hakkında bilgi verir. Çevre modeli, çevrenin nasıl davranacağı konusunda çıkarımlar yapılmasına olanak sağlayan bir yapıdır.

Pekiştirmeli öğrenme yöntemleri şu şekilde verilebilir:

- ÇKS
- Sonlu Markov Karar Süreçleri (Finite Markov Decision Process)
  - Dinamik Programlama (Dynamic Programming)
  - Monte Carlo Yöntemleri (Monte Carlo Methods)
  - Geçici-Fark Öğrenme (Temporal-difference Learning)

### 2.3.1. Çok kollu slot

Çeşitli işletmelerde farklı amaçlar için kullanılan çalışması için bir adet kolu bulunan makinelere slot makinesi denmektedir. Bu makinelerin çalışması için makinede bulunan kolun çekilmesi gerekmektedir. Kolun çekilmesinden sonra slot makinesi çalışmaya başlar. Çalışma sonucunda bir ödül alınır veya alınmaz. ÇKS çalışma metodu da aslında slot makinesine benzerdir. Tek farkı bir adet değil  $k$  adet slot makinesi ve her makinenin ayrı bir kolu bulunmaktadır. Her koldan elde edilecek ödülün dağılımı belli değildir.

ÇKS,  $k$ -kollu slot problemi olarak da isimlendirilmektedir.  $K$ , kol sayısını yani seçebileceğimiz durumları belirtir. Her bir kol seçimi sonucunda ödül alınır veya alınmaz. Alınan ödül değerine göre bir dahaki kol seçiminin nasıl yapılacağına karar verir (Agrawal ve Goyal, 2012). Hedef, belirli bir süre için çekilen kollardan elde edilen beklenen toplam ödülü en üst seviyeye çıkarmaktır. Yukarıda açıklanan ÇKS mekanizması şu şekilde özetlenebilir:

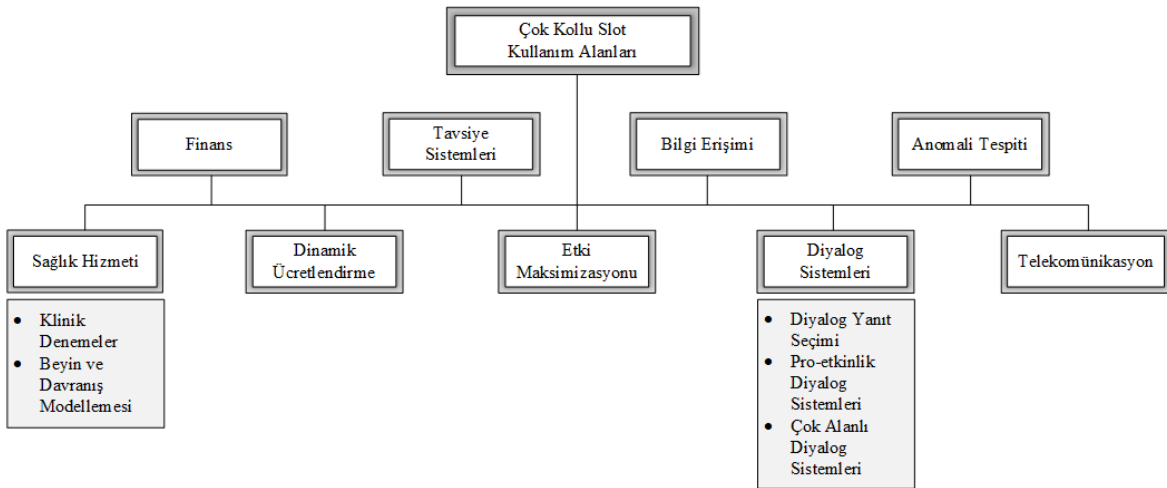
- $A_t$ :  $t$  zaman adımında seçilen aksiyon
- $t$ : zaman adımı
- $R_t$ :  $A_t$  sonucunda elde edilen ödül
- $a$ : seçilen aksiyonun değeri
- $q_*(a)$ :  $a$ 'nın seçilmesi durumunda beklenen ödül

$$q_*(a) \doteq E[R_t | A_t = a] \quad (2.1)$$

- $Q_t(a)$ :  $t$  zaman adımında  $a$ 'nın seçilmesi durumunda tahmin edilen değer

Her bir zaman adımında aksiyonların tahmini değerleri içerisinde tahmin değeri en büyük olan en az bir tane eylem vardır. Bu eylemlere aç gözlü (greedy) eylemler denir. Şu ana kadar edinilmiş bilgiden yararlanarak en fazla değeri veren aç gözlü bir eylem seçilirse sömürme, ancak aç gözlü yerine gelecekte daha yararlı olabilecek diğer eylemlerden seçilirse keşif gerçekleştirilmiş olmaktadır. Sömürü ile bir zaman adımında beklenen ödül en üst seviyeye çıkarılırken keşif ile uzun vadede toplam ödül en üst seviyeye çıkarılabilir (Sutton ve Barto, 2018). Bu durum keşfetme sömürü ikilemi (dengesi) olarak bilinmektedir. Ayrıca ÇKS, çevrimiçi öğrenmede kullanılan yöntemlerden biridir (Hoi vd., 2018). Çevrimiçi öğrenme yöntemleri geleneksel öğrenme yöntemlerinden farklı olarak öğrenmeyi her bir yeni veri geldiğinde sürekli olarak devam ettirmektedir. ÇKS her zaman adımında öğrenmeyi devam ettiren yapısı nedeniyle çevrimiçi öğrenme yöntemlerinden biridir.

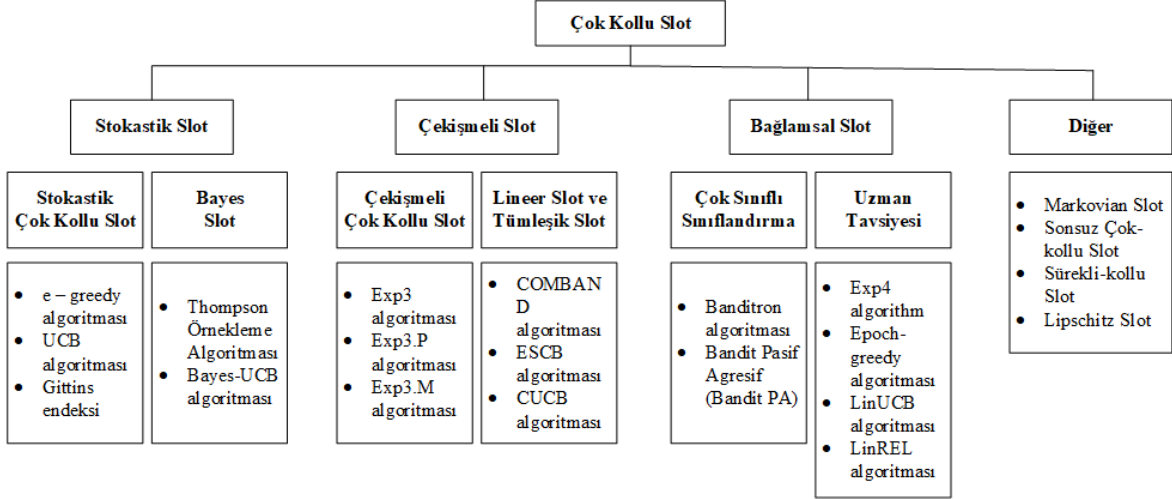
ÇKS, birçok farklı uygulama alanında kullanılmaktadır. Literatürde ÇKS kullanarak; anomali tespiti (Ding vd., 2019), finans (Shen vd., 2015), sağlık (Zhou vd., 2020), öneri sistemleri (Barraza-Urbina ve Glowacka, 2020), bilgi erişimi (Losada vd., 2017) alanlarında çalışmalar bulunmaktadır. Şekil 2.7’de ÇKS kullanım alanları daha detaylı olarak verilmektedir.



Şekil 2.7. ÇKS kullanım alanları (Bouneffouf ve Rish, 2019)

ÇKS kapsamında birçok farklı uygulama alanı olduğu gibi bu alanlarda kullanılan farklı yöntemler de bulunmaktadır. Bu yöntemlerin incelenmesi için literatürde ÇKS üzerine yapılmış çalışmalar bulunmaktadır (Hoi vd., 2018). Yapılan bu çalışmalarda ÇKS

kapsamında kullanılan yöntem ve algoritmalar incelenmiş ve Şekil 2.8’de verilen taksonomi oluşturulmuştur.



Şekil 2.8. ÇKS taksonomisi (Hoi vd., 2018)

Görüldüğü üzere farklı özelliklere sahip ÇKS algoritmaları bulunmaktadır. Bu çalışmada “Upper Confidence Bounds (UCB)” algoritmasına odaklanılmıştır. Devam eden bölümde UCB hakkında genel temel bilgiler verilmektedir.

### 2.3.1.1. UCB algoritması

UCB algoritması stokastik ÇKS içerisinde bulunmaktadır (Bkz. Şekil 2.8). UCB politikası Agrawal’ın (1995) indeks tabanlı politikasından türetilmiştir (Auer vd., 2002). UCB, belirsizlik karşısında iyimserlik ilkesini kullanarak keşif ve sömürüyü eşzamanlı olarak gerçekleştiren bir stratejidir (Hoi vd., 2018). UCB, hangi eylemlerin en iyisi olduğuna dair bilgi olmadan, her eylemin beklenen getirisinin/kaybının ne kadar iyi olduğuna dair iyimser bir tahmin oluşturur ve en iyi tahminle eylemi seçer. Eğer eylem seçimi doğru ise bu seçime güvenilerek bir dahaki seçimlerde de bu eylem seçimine devam edebilir ve bu nedenle pişmanlık duyabiliriz. Bu sömürü olarak tanımlanabilir. Sömürü sonucu yapılan eylem seçimi her zaman doğru eylem seçimi olmayabilir. Böyle bir durumda da yanlış eylem seçimden dolayı iyimser tahmin azalır ve bu nedenle de farklı bir eylem seçimi yapmaya başlanacaktır. Bu keşif olarak tanımlanabilir. Bu şekilde keşif-sömürü dengesi sağlanır.

Yukarıda bahsedilen eylem seçiminde kullanılan “iyimserlik tahmini”, UCB belirtmektedir. UCB formülü şu şekildedir (Roberts, 2020):

$$A_t = \operatorname{argmax}_a \left[ Q_t(a) + c \sqrt{\frac{\log t}{N_t(a)}} \right] \quad (2.2)$$

Formülde yer alan parametreler (Roberts, 2020):

- $A_t$ : " $t$ " adımımda seçilen eylemdir.
- $Q_t(a)$ : " $t$ " adımımdaki " $a$ " eyleminin tahmini değeridir.
- $N_t(a)$ : " $t$ " zamanından önce " $a$ " eyleminin seçilme sayısıdır.
- $c$ : keşif düzeyini kontrol eden bir güven değeridir.

UCB formülünün 2 farklı bölümden oluştuğu düşünülebilir:

- Sömürü:  $Q_t(a)$  denklemin sömürü kısmını temsil eder.
- Keşif: Denklemin ikinci yarısı keşif kısmını temsil eder. Keşif derecesi “ $c$ ” hiper parametresi tarafından kontrol edilir.

### 2.3.2. Çok kollu slot ve anomali tespiti

ÇKS algoritmaları literatürde birçok alanda, farklı amaçlar için kullanılmaktadır (Bkz. Şekil 2.7). Anomali tespiti de bu amaçlardan biridir. Literatürde ÇKS kullanılarak yapılan anomali tespiti çalışmaları bulunmaktadır (Zhuang vd., 2017; Ding vd., 2019; Prabhu vd., 2019). Zhuang vd. (2017), çok kollu bir slottaki diğer kollara kıyasla son derece yüksek / düşük ödül beklentileri olan aykırı/anormal kolları belirlemek için Round-Robin algoritması ve Ağırlıklı Round-Robin algoritması önermişlerdir. Ding vd. (2019), sistemin gerçek anomalileri hakkında insan uzmanla proaktif olarak iletişim kurarak, etkileşimli bir ortamda ilişkilendirilen ağlarda anormallik algılama için yeni bir bağlamsal ÇKS algoritması “GraphUCB” sunmuşlardır.

Endüstri 4.0 ile birlikte gelişen akıllı fabrika konsepti beraberinde endüstriyel sistemlerde OTA kullanımını artmaya başlamıştır. OTA endüstride taşıma görevlerini yerine getirmektedir. OTA’da meydana gelebilecek ortam veya OTA kaynaklı anomali tüm

sistemin çalışmasını ve güvenliği olumsuz etkileyebilir. Bu nedenle OTA için anormal durumların tespit edilmesi önemlidir. Ancak OTA için anormal durumların neler olabileceği önceden belirlenemeyebilir. Bu nedenle normal ve anormal durumların öğrenilebilmesi yani çevrim içi bir öğrenme ile anomali tespiti gerçekleştirilmesi gerekebilir. Bu duruma uyum sağlayabilecek öğrenme yöntemlerine örnek olarak pekiştirmeli öğrenme olan ÇKS algoritmaları verilebilir. Ferdowsi vd. (2019), otonom bağlantılı araçların sensörlerine yönelik siber saldırıları ve bunların sistemi üzerindeki fiziksel etkilerini ele almak için veri enjeksiyon saldırısı algılama yaklaşımı önermişlerdir. Önerilen yaklaşım sensör okumalarının stokastik davranışından yararlanmaya ve ÇKS kullanımına dayanmaktadır. ÇKS altında bulunan UCB algoritması kullanılarak belirlenen sensör alt kümelerinden en güvenilir sensör alt kümesi seçiminin gerçekleştirmiş ve seçilen alt kümede bulunmayan sensör için anormallik bulunduğu karar vermişlerdir. Değirmenci vd. (2020), pekiştirmeli öğrenme kapsamında “LinUCB” algoritmasını kullanarak akıllı fabrikalarda OTA’ın kullandığı kavşakların karakteristiğini öğrenmişlerdir. Kavşak karakteristiklerinde değişim olması durumunu anormal durum olarak tespit etmişlerdir.

#### **2.4. İstatiksel Proses Kontrol**

İPK, proseslerde oluşabilecek değişkenliğin azaltılması yoluyla proses verimliliği ve kararlılığının sağlanmasında faydalı olan güçlü bir problem çözme araçları koleksiyonudur (Montgomery, 2009). İPK kullanıldığı alanlara örnek olarak üretim yapılan sistemler verilebilir.

Üretim süreçlerinde belirli bir miktarda doğal değişkenlik bulunmaktadır. Bu doğal değişkenlik kararlı sistemin “tesadüfi nedeni (chance causes)” olarak adlandırılır. Proseste sadece tesadüfi neden nedeniyle bir değişkenlik var ise prosesin içsel parçası olarak kabul edilir ve proses istatistiksel kontrolde olarak tanımlanır (Montgomery, 2009). Tanımlanan doğal değişkenlik harici proseste başka çeşit değişkenliklerde meydana gelebilir. Tesadüfi neden modelinin parçası olmayan bu değişkenlik kaynakları, “atanabilir varyasyon nedenleri (assignable causes of variation)” olarak tanımlanmaktadır (Montgomery, 2009). Proseste atanabilir nedenlerin olması durumunda proses kontrol dışı olarak kabul edilir. İPK amacı da proseste oluşabilecek atanabilir nedenleri tespit etmek ve sorunu araştırarak

durumu düzeltmeye çalışmaktır. Sonuç olarak İPK amacı prosesdeki değişkenliğin tespit edilmesi ve değişkenlik sorununa çözüm sağlamak olduğu söylenebilir.

#### **2.4.1. İstatiksel proses kontrol çeşitleri**

İPK, proseslerin takibinde kullanılmaktadır. Proseslerin takibi ve proseslerde oluşabilecek anormal durumların tespit edilmesi için yedi farklı yöntem verilebilir (Montgomery, 2009). Bunlar:

- Histogram
- Kontrol Sayfası (Check Sheet)
- Pareto Grafiği
- Neden ve Etki Diyagramı
- Kusur Konsantrasyon Diyagramı
- Dağılım Diyagramı ve
- Kontrol Grafiğidir.

İPK çeşitlerinde farklı yöntemler kullanılmaktadır ancak genel amaç proseslerdeki değişiklik, anormal durumların gözlenebilmesi ve prosesin kontrol durumunu koruyabilmektir. Devam eden bölümde kontrol grafikleri için bilgiler verilmektedir.

#### **2.4.2. Kontrol grafikleri**

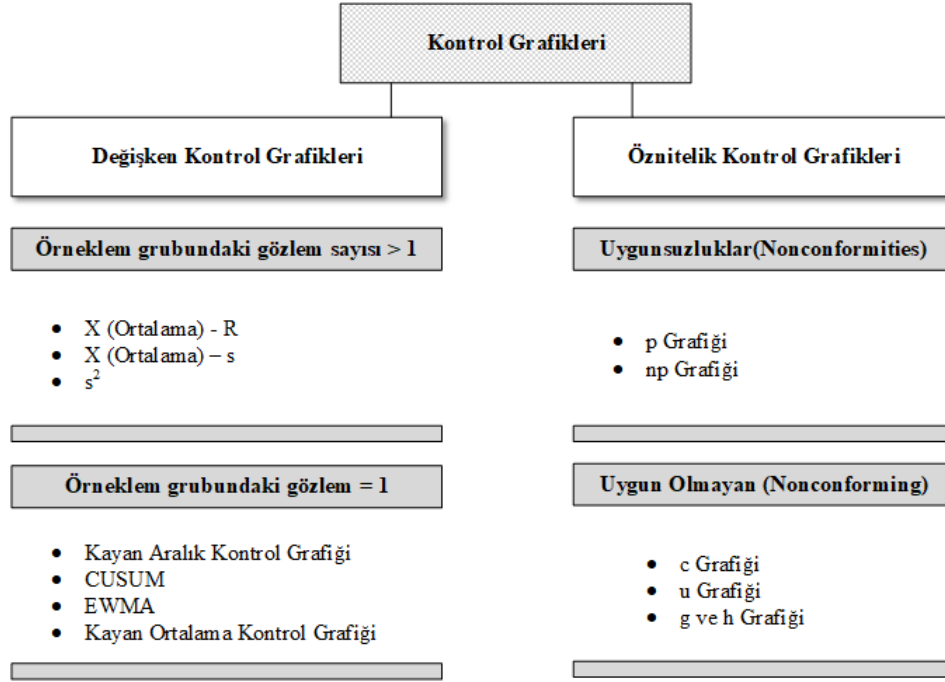
Kontrol grafikleri, İPK'de kullanılan yöntemlerden biridir. Kontrol grafiklerinin oluşturulması için bazı temel işlemler bulunmaktadır. Bunlar:

- Merkez çizginin belirlenmesi: Grafiğin merkez çizgisini belirtir.
- Üst Kontrol Limitinin (ÜKL) belirlenmesi: Grafik değerlerinin olası çıkabileceği en yüksek değeri belirtir.
- Alt Kontrol Limitinin (AKL) belirlenmesi: Grafik değerlerinin olası düşebileceği en düşük değeri belirtir.
- Kurallar: Üst ve alt kontrol limitleri arasında veya üstünde veri noktalarının izlediği davranışa göre belirlenir.



- Alt Gruplar: Grafik çiziminde kullanılacak veri setinde bulunan verilerin belli sayıda veri içeren gruplara ayrılmasıdır.
- Örnek Boyutu: Her bir alt grup içerisinde kaç adet veri olacağıdır.
- Örnek Frekansı: Veri setinde bulunan verilerin üretilme veya alınma sıklığı olarak verilebilir.

Yukarıda verilen işlemlerin gerçekleştirilmesi kontrol grafiğinde kullanılacak verinin kesikli veya kesiksiz olması, kesiksiz veride örnek boyutunun büyüklüğü gibi durumlara göre farklılaşmaktadır. Şekil 2.9'da kontrol grafiklerinin çeşitleri bu bakış açısından verilmektedir.



Şekil 2.9. Kontrol grafiği çeşitleri

Değişken kontrol grafikleri, veri kümesinin kesiksiz değerler içermesi durumunda kullanılmaktadır. Kendi içerisinde alt grup içerisinde bulunan örnek boyutuna göre ikiye ayrılmaktadır. Örnek boyutu birden büyük olduğu durumlarda  $x$  ve  $r$ ,  $x$  ve  $s$  ve  $s^2$  grafikleri kullanılabilirken bire eşit olduğu durumlarda ise kayan aralık, CUSUM, EWMA ve kayan ortalama kontrol grafikleri kullanılabilir (Montgomery, 2009).

Öz nitelik kontrol grafikleri ise, veri setinin kesikli değerler içermesi durumunda kullanılmaktadır. İkiye ayrılmaktadır. Bunlar; “uygun olmayan (nonconforming)” ve “uygunsuzluklar (nonconformities)” olarak verilebilir. Bu kategoriler altında kullanılan yöntemler Şekil 2.9’da görülmektedir. Bu çalışmada odak nokta olarak değişken kontrol grafikleri belirlenmiştir.

Kontrol grafiklerinin asıl amacı olan proseslerdeki anormal durumların tespit edilmesi iki aşamalıdır. Bu aşamalar faz 1 ve faz 2 olarak adlandırılmaktadır. Faz 1, kontrol grafiği çizilecek prosesin alınan verilerinde grafiğin oluşturulmasında hataya yol açabilecek durumların tespit edilmesi veya sorun olmadığının doğrulanması aşaması olarak açıklanabilir. Faz 1’de:

- Veri toplanır: İzlenecek prosesin grafiğinin oluşturulması için prosesin normal dağılım gösterdiği belirlenen zaman aralığından veriler elde edilir.
- Kontrol limitleri (merkez çizgi, ÜKL ve AKL) belirlenir: Toplanan veriler belirlenen kontrol grafiği tipine göre işlenerek kontrol limitleri oluşturulur.
- Verinin kontrol limitlerine uygunluğu analiz edilir: Oluşturulan kontrol limitleri ile ilk aşamada toplanan verilerin grafiği oluşturularak verilerin belirlenen alt ve üst limitler aralığında bulunup bulunmadığı kontrol edilir. Eğer veriler belirlenen aralıkta bulunuyorsa birinci faz başarılıdır. Ancak tüm veriler aralıkta bulunmuyor ise limitler dışına çıkan veriler tekrar incelenir, normal dağılımı bozan gerekli görülmeyen durumlar veriler içerisinde çıkarılarak kontrol limitleri tekrar belirlenir. Bu işlemler başarılı sonuç elde edilene kadar devam eder.

Faz 2, faz 1 sonucunda izlenecek prosesin kontrol grafiğine ait değerlerinin tamamlanmasının ardından gerçek zamanlı olarak prosesten gelen verilerin izlenmesidir. Bu aşamada gelen verilerin kontrol limitleri arasında bulunup bulunmadığı kontrol edilir. Eğer herhangi bir veri kontrol limitleri dışına çıkar ise bu durum aykırı (anormal) durum olarak tespit edilir ve sistem “kontrol dışı” olarak adlandırılır. Tüm bu işlemlerin gerçekleştirilmesinde büyük önemi olan bileşen, kontrol limitlerinin belirlenmesidir. Değişken kontrol grafikleri olan  $x$  ve  $r$ ,  $\bar{x}$  ve  $s$ , kayan aralık grafiklerine ait formüller (Montgomery, 2009) Çizelge 2.1’de verilmektedir. Montgomery (2009), formüllerde kullanılan sabit değerlerin elde edilmesi ilgili detaylı bilgi vermektedir.

Çizelge 2.1. Değişken kontrol grafikleri kontrol limiti hesaplanma formülleri ve açıklamaları (Montgomery, 2009)

Değişken Kontrol Grafik Tipleri	Açıklama	Formül
<b>Sabit standart sapma ve ortalama değerinin verildiği</b>		
$\bar{x}$ Grafiği	$\mu$ ve $\sigma$ : ortalama, standart sapma değeri	Merkez çizgisi: $\mu$ ÜKL, AKL: $\mu \pm A\sigma$ (2.3)
$R$	$\sigma$ : standart sapma değeri $d_2$ : sabit $D_2, D_1$ : sabit	Merkez çizgisi: $d_2\sigma$ ÜKL = $D_2\sigma$ AKL = $D_1\sigma$ (2.4)
$s$	$\sigma$ : standart sapma değeri $B_5, B_6$ : sabit $c_4$ : sabit	Merkez çizgisi: $c_4\sigma$ ÜKL = $B_6\sigma$ AKL = $B_5\sigma$ (2.5)
<b>Sabit standart sapma ve ortalama değeri verilmediği geçmiş verinin kullanıldığı</b>		
$\bar{x}$ ve $R$ grafiği	$\bar{x}$ : grup içerisinde bulunan verilerin ortalaması $\bar{\bar{x}}$ : her grup için hesaplanan $\bar{x}$ değerlerinin ortalaması $R$ : grup içerisindeki en büyük ve en küçük verinin farkı $\bar{R}$ : her grup için hesaplanan $R$ değerlerinin ortalaması $A_2$ : sabit (grup içerisindeki veriye göre değişiklik gösterir.)	Merkez çizgisi: $\bar{\bar{x}}$ ÜKL = $\bar{\bar{x}} + A_2\bar{R}$ AKL = $\bar{\bar{x}} - A_2\bar{R}$ (2.6)
$R$ grafiği	$R$ : grup içerisindeki en büyük ve en küçük verinin farkı $\bar{R}$ : her grup için hesaplanan $R$ değerlerinin ortalaması $D_3$ ve $D_4$ : sabit (grup içerisindeki veriye göre değişiklik gösterir.)	Merkez çizgisi: $\bar{R}$ ÜKL = $D_4\bar{R}$ AKL = $D_3\bar{R}$ (2.7)

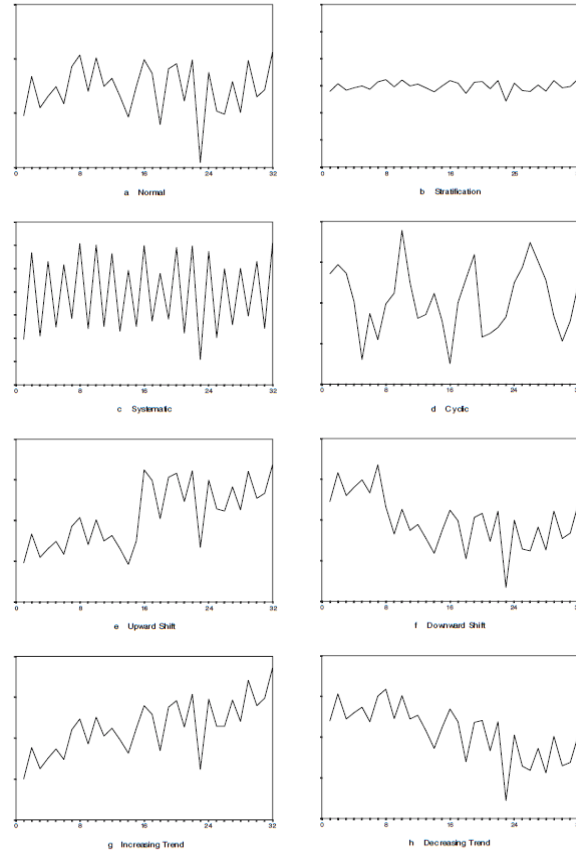
Çizelge 2.1. Değişken kontrol grafikleri kontrol limiti hesaplanma formülleri ve açıklamaları (Montgomery, 2009) (devam)

$\bar{x}$ ve s grafiği	$\bar{x}$ : grup içerisinde bulunan verilerin ortalaması $\bar{\bar{x}}$ : her grup için hesaplanan x- değerlerinin ortalaması s: her gruba ait hesaplanma standart sapma değeri $\bar{s}$ : her gruba ait s değerlerinin ortalaması $A_3$ : sabit (grup içerisindeki veriye göre değişiklik gösterir.)	Merkez çizgisi: $\bar{\bar{x}}$ ÜKL = $\bar{\bar{x}} + A_3\bar{s}$ AKL = $\bar{\bar{x}} - A_3\bar{s}$ (2.8)
s grafiği	s: her gruba ait hesaplanma standart sapma değeri $\bar{s}$ : her gruba ait s değerlerinin ortalaması $B_3$ ve $B_4$ : sabit (grup içerisindeki veriye göre değişiklik gösterir.)	Merkez çizgisi: $\bar{s}$ ÜKL = $B_4\bar{s}$ AKL = $B_3\bar{s}$ (2.9)
Kayan aralık kontrol grafiği	$MR$ : her iki verinin birbirinden farkının kayarak hesaplanması ve bu değerlerin toplamı $\overline{MR}$ : MR değerinin ortalaması $d_2$ : sabit $\bar{x}$ : tüm verilerin ortalaması	Merkez çizgisi: $\bar{x}$ ÜKL = $\bar{x} + 3 * \frac{\overline{MR}}{d_2}$ AKL = $\bar{x} - 3 * \frac{\overline{MR}}{d_2}$ (2.10)
Kayan aralık kontrol grafiği (opsiyonel)	$MR$ : her iki verinin birbirinden farkının kayarak hesaplanması ve bu değerlerin toplamı $\overline{MR}$ : MR değerinin ortalaması $D_3$ ve $D_4$ : sabit ( $D_3 = 0, D_4 = 3.267$ )	Merkez çizgisi: $\overline{MR}$ ÜKL = $D_4\overline{MR}$ AKL = $D_3\overline{MR}$ (2.11)

Kontrol grafiklerinin limitleri seçilen kontrol grafiği çeşidine göre belirlenir ve sistem belirlenen limitlere göre takip edilir. Ancak bazı durumlarda daha iyi bir kontrol ve analiz için bir kere belirlenip hep kullanılan limitler yerine dinamik olarak limitlerin elde edilmesi gerekebilir. Literatürde bu amaçla yapılmış çalışmalar bulunmaktadır (Parkhideh ve Case, 1989; Wasserman, 1994; Song ve Vorburger, 2007; Kovářík ve Klímek, 2012; Liang vd., 2017). Yapılan çalışmalarda var olan kontrol grafiklerinin geliştirilmesi ve yeni kontrol grafiklerinin oluşturulması yaygındır. Song ve Vorburger (2007) ise bu durumdan farklı olarak kontrol limitlerinde genellikle kullanılan sabit ortalama ve standart sapma değerlerinin dinamik olarak güncelleştirilmesini önermiştir.

### 2.4.3. Kontrol grafiklerinde örüntü tanıma

Kontrol grafikleri üretim aşamalarının izlenmesi ve kontrol edilmesi için yaygın olarak kullanılmaktadır. Ancak kontrol grafiklerinin kullanıldığı sistemlerin verimliliği için grafik örüntüsünde oluşabilecek kontrol dışı durumların tespit edilebilmesi de önemlidir (Fuqua ve Razzaghi, 2020; Lu vd., 2020). Örüntüler normal veya anormal olarak sınıflandırılabilir. Kontrol grafiklerinin oluşturulmasında genel olarak normal dağılıma, örüntüye, sahip veriler kullanıldığı için normal örüntü prosesin kontrol altında olduğunu temsil ederken normal örüntü haricindeki durumlar anormal, kontrol dışı durum olarak kabul edilebilir. Kontrol grafiği örüntüsü (Control Chart Pattern, CCP) çeşitlerini Gauri ve Chakraborty (2007) sekiz adet (normal, katmanlaşma, sistematik, döngüsel, yukarı kayma, aşağı kayma, yukarı trend ve aşağı trend) tanımlamıştır. Başka çalışmalarda farklı sayıda tanımlamalar da bulunmaktadır (Pham ve Wani, 1997; Wani ve Rashid, 2005). Şekil 2.10'da sekiz adet CCP çeşidi örneği verilmektedir.



Şekil 2.10. CCP çeşitleri: (a) normal, (b) katmanlaşma, (c) sistematik, (d) döngüsel, (e) yukarı kayma, (f) aşağı kayma, (g) artan trend, (h) azalan trend (Gauri ve Chakraborty'dan, 2007)

Trend, kayma, sistematik ve döngüsel örüntüler şu şekilde tanımlanabilir (Hachicha ve Ghorbel, 2012):

- Trend: yukarı veya aşağı yönde sürekli bir ilerleme
- Kayma: prosesin ortalamasındaki ani değişiklikler
- Sistematik: Düşük değerli bir nokta her zaman yüksek değerli bir nokta takip eder. Bu durum normal örüntüde tahmin edilemez, sistematik değildir.
- Döngüsel: Alçak kısımlar veya çukurlar serpiştirilmiş bir dizi yüksek kısım veya tepe ile tanınabilir.

CCP tanımında kullanılacak yöntemden önce gelen bir önemli konuda kullanılacak verilerdir. CCP tanımında iki çeşit veri kullanılır:

- Gerçek sistemden elde edilen veri veya
- Simülasyon ile elde edilen veri

Gerçek sistemden istenilen CCP çeşitlerinde verilerin elde edilmesi zordur. Bu nedenle çoğu çalışmada simülasyon ile CCP çeşitleri için veri üretilerek CCP tanımında kullanılmaktadır. CCP çeşitlerinin üretilmesi için kullanılacak formüller Çizelge 2.2’de verilmektedir (Gauri ve Chakraborty, 2007).

Çizelge 2.2. CCP Çeşitlerinin üretiminde kullanılacak formüller (Gauri ve Chakraborty, 2007)

CCP Çeşidi	Formül
Normal	$y_i = \mu + r_i\sigma$ (2.12)
Katmanlaşma	$y_i = \mu + r_i\sigma'$ (2.13)
Sistematik	$y_i = \mu + r_i\sigma + d * (-1)^i$ (2.14)
Yukarı veya Aşağı Trend	$y_i = \mu + r_i\sigma \pm ig$ (2.15)
Yukarı veya Aşağı Kayma	$y_i = \mu + r_i\sigma \pm ks$ (2.16)
Döngüsel	$y_i = \mu + r_i\sigma + a \sin\left(\frac{2\pi i}{T}\right)$ (2.17)

Çizelge 2.2’de verilen formüllerde kullanılan sembollerin açıklamaları şu şekildedir (Gauri ve Chakraborty, 2007):

- $\mu$ : prosesin ortalama değeri
- $\sigma$ : prosesin standart sapma değeri

- $\sigma'$ : katmanlaşma örüntüsü için rastgele gürültü değeri (Örnek: 0.2, 0.3 veya  $0.4 \cdot \sigma$  seçilebilir)
- $a$ : dögüsel varyasyonun genliđi (Örnek: 1.5, 2.0 veya  $2.5 \cdot \sigma$  seçilebilir)
- $g$ : trend örüntüsü için gradyan büyüklüğü (Örnek: 0.05, 0.075 veya  $0.10 \cdot \sigma$  seçilebilir)
- $d$ : sistematik örüntüsünün büyüklüğü (Örnek: 1, 2 veya  $3 \cdot \sigma$  seçilebilir)
- $k$ : kayma konumunu belirleyen parametre (Örnek: pencere genişliđi 32 için 9, 17 ve 25 olarak seçilebilir)
- $s$ : kayma örüntüsünün büyüklüğü (Örnek: 1.5, 2.0 veya  $2.5 \cdot \sigma$  seçilebilir)
- $i$ : örüntünün örneklendiđi kesikli zaman noktası (Örnek: pencere genişliđi 32 için 1 ila 32 aralıđında alınır)
- $T$ : dögü periyodu (Örnek: pencere genişliđi 32 için 8 ve 16 örnekleme aralıđı olarak seçilebilir)
- $r_i$ : standart bir normal deđişkenin rastgele değeri
- $y_i$ :  $i$ 'inci zaman noktasındaki örnek deđer

İlk CCP tanıma yaklaşımı 1987 yılında J.A. Swift (Swift, 1987) tarafından geliştirilmiştir (Hachicha ve Ghorbel, 2012). Çalışmada ortalama ve varyans ani deđişimlerini (kayma) sezgisel kurallar ve istatikselsel testler kullanarak tespit edilmiştir. CCP tanınması için birçok farklı yöntem kullanılmıştır. Zone testleri, run testleri ve geometrik kayan ortalama testi bu yöntemlerdendir. Bu yöntemler anormal bir örüntünün oluştuđunu tespit edebilir ancak hangi örüntünün oluştuđunu net olarak belirtememektedir (Hachicha ve Ghorbel, 2012). Bu nedenle araştırmacılar daha farklı yöntemler geliştirmiş ve kullanmışlardır. Literatürde CCP tanıma için kullanılan yöntemler genel olarak şu şekilde verilebilir (Gauri ve Chakraborty, 2007):

- Uzman Sistemler / Kural Tabanlı (Experts Systems/Rule-based): Uzman kişiler tarafından oluşturulan kurallara göre CCP tanıma yapılır.
- Sinir Ağları: Sinir ağları algoritması tabanlı uygulamalar ile CCP tanıma yapılır
- Matematiksel Örüntü Özellikleri: İstatikselsel özellikler kullanılarak CCP tanıma yapılır
- Öznitelik-tabanlı Sezgisel: Öznitelikler yani veriler kullanılarak yapılan hesaplamalar ile CCP tanıma yapılır

Öznitelik-tabanlı yöntemlerde ham veriler kullanılarak yapılan hesaplamalar sonucu elde edilen öznitelikler ile CCP tanıma yapılır. İki aşamalıdır. İlk aşamada veriler kullanılarak yapılan hesaplamalar sonucunda öznitelikler elde edilir. İkinci aşamada ise elde edilen öznitelikler kullanılarak CCP tanıma gerçekleştirilir. CCP tanıma için çalışmalarda farklı sayıda öznitelik kullanılmıştır. Pham ve Wani (1997) dokuz adet, Gauri ve Chakraborty (2007) 32 adet, Ebrahimzadeh ve Ranaee (2009) yedi adet, Wang vd. (2009) dokuz adet ve Gauri (2010) yedi adet öznitelik kullanmıştır.

Çevrimiçi CCP tanıma için kayan pencere yaklaşımı (Guh, 2008; Guh, 2010) önemlidir (Hachicha ve Ghorbel, 2012). Öznitelik-tabanlı yöntemlerde de pencere yöntemi uygulanmaktadır (Gauri ve Chakraborty, 2007). Özniteliklerin hesaplanması için veriler belirlenen pencere (window) genişliği kadar veri içeren gruplara ayrılır. Her pencere genişliği kaydırıldığında içinde yine aynı miktarda veri bulunur. Kaydırma büyüklüğü çalışmaya göre değişebilir. Pencere genişliğinin belirlenmesi önemli bir konudur. Büyük pencere genişliği tanıma, tespit için gereken süreyi arttırarak olumsuz etki oluşturabilir (Gauri ve Chakraborty, 2007). Örnek olarak Gauri ve Chakraborty (2007) çalışmalarında pencere genişliğini 32 olarak seçmiştir. Bazı özniteliklerin hesaplanmasında belirlenen pencere genişliği de dört veya iki gruba bölünür ve hesaplamalar gerçekleştirilir. Çalışmalarda kullanılan bazı öznitelikler ve açıklamaları şu şekildedir:

- Eğim ( $B$ ): Örüntü (pencere genişliğindeki veri) en küçük kare (Least squares, LS) çizgisine ait eğim
- $RVE$ : Örüntünün varyansı ile örüntünün LS çizgisinin ortalama hata karelerinin toplamı arasındaki oran
- $RDIST$ : SS açısından ardışık noktalar arasındaki ortalama mesafe
- $AASBP$ : SS açısından ardışık noktalardan geçen düz çizgilerin ortalama mutlak eğimi
- $nc2$ : LS çizgisini geçen örüntü içerisindeki veri sayısı
- $nlc$ : Ortalama çizgisini geçen örüntü içerisindeki veri sayısı
- $cmember$ : Örüntünün ne kadar döngüsel örüntüye benzediğini belirtir
- $a_{pm}$ : Örüntü ve ortalama çizgisi arasındaki alan
- $a_{ps}$ : Örüntü ve LS çizgisi arasındaki alan
- $a_{ss}$ : Örüntü ve örüntüye oturtulmuş LS çizgisi arasındaki alan



- *skew*: Çarpıklık. Örüntünün asimetri bilgisini verir
- *kurtosis*: Basıklık. Örüntünün dağılımının nispi zirvesini veya düzlüğünü verir
- $a_s$ : Çizgi parçalarının LS çizgilerinin ortalama eğimi
- *sd*: Örüntüye ait LS çizgisi eğimi ile iki pencere içindeki örüntülere ait çizgi parçalarının ortalama eğimi arasındaki fark

Yukarıda açıklamaları verilen bazı özniteliklere ek olarak kullanılan diğer özniteliklere ait formül ve açıklamalar Çizelge 2.3’de verilmektedir.

Çizelge 2.3. CCP tanıma için kullanılan bazı özniteliklerin adı, formül açıklaması ve formülü

Öznitelik Adı	Açıklaması	Formülü
<i>ACLPI</i>	<i>N</i> : örüntü içindeki veri sayısı <i>ACL</i> : örüntü ve ortalama çizgisi arasındaki alan <i>SD</i> : örüntünün standart sapması	$ACLPI = \left[ \frac{ACL}{(N-1)} \right] / SD$ (2.18)
<i>ACLMLC</i>	<i>ACL</i> : örüntü ve ortalama çizgisi arasındaki alan <i>MLC</i> : ortalama çizgisini geçen örüntü içerisindeki veri sayısı <i>SD</i> : örüntünün standart sapması	$ACLMLC = \left[ \frac{ACL}{MLC} \right] / SD$ (2.19)
<i>ALSPI</i>	<i>ALS</i> : örüntü ve örüntüye oturtulmuş LS çizgisi arasındaki alan <i>N</i> : örüntü içindeki veri sayısı <i>SD</i> : örüntünün standart sapması	$ALSPI = \left[ \frac{ALS}{N-1} \right] / SD$ (2.20)
<i>ALSLSC</i>	<i>ALS</i> : örüntü ve örüntüye oturtulmuş LS çizgisi arasındaki alan <i>LSC</i> : LS çizgisini geçen örüntü içerisindeki veri sayısı <i>SD</i> : örüntünün standart sapması	$ALSLSC = \left[ \frac{ALS}{LSC} \right] / SD$ (2.21)
<i>RACLALS</i>	<i>ACL</i> (örüntü ve ortalama çizgisi arasındaki alan) ve <i>ALS</i> (örüntü ve örüntüye oturtulmuş LS çizgisi arasındaki alan) oranı	$RACLALS = \frac{ACL}{ALS}$ (2.22)

Çizelge 2.3. CCP tanıma için kullanılan bazı özneliklerin adı, formül açıklaması ve formülü (devam)

<i>PMLC</i>	<i>MLC</i> : ortalama çizgisini geçen örüntü içerisindeki veri sayısı <i>N</i> : örüntü içindeki veri sayısı	$PMLC = \frac{MLC}{N}$ (2.23)
<i>PLSC</i>	<i>LSC</i> : LS çizgisini geçen örüntü içerisindeki veri sayısı <i>N</i> : örüntü içindeki veri sayısı	$PLSC = \frac{LSC}{N}$ (2.24)
<i>PSMLSC</i>	<i>MLC</i> : ortalama çizgisini geçen örüntü içerisindeki veri sayısı <i>LSC</i> : LS çizgisini geçen örüntü içerisindeki veri sayısı <i>N</i> : örüntü içindeki veri sayısı	$PSMLSC = \frac{MLC+LSC}{2N}$ (2.25)
<i>PSMLSC</i>	$o_i$ : ortalama çizgisini geçen örüntü içerisindeki veri sayısı $o_i'$ : LS çizgisini geçen örüntü içerisindeki veri sayısı <i>N</i> : örüntü içindeki veri sayısı	$PSMLSC = \sum_{i=1}^{N-1} \frac{(o_i + o_i')}{2(N-1)}$ (2.26)
<i>skew1</i>	$\mu$ : örüntünün ortalama değeri <i>N</i> : örüntüdeki veri sayısı $x_i$ : örüntüdeki <i>i</i> . Veri $\sigma$ : örüntünün standart sapması	$skew1 = \frac{\sum_{i=1}^N (x_i - \mu)^3}{\sigma^3}$ (2.27)
<i>kurtosis1</i>	$\mu$ : örüntünün ortalama değeri <i>N</i> : örüntüdeki veri sayısı $x_i$ : örüntüdeki <i>i</i> . Veri $\sigma$ : örüntünün standart sapması	$kurtosis1 = \frac{\sum_{i=1}^N (x_i - \mu)^4}{\sigma^4}$ (2.28)
<i>skew2</i>	$\mu$ : örüntünün ortalama değeri <i>N</i> : örüntüdeki veri sayısı $x_i$ : örüntüdeki <i>i</i> . Veri $\sigma$ : örüntünün standart sapması	$skew2 = \frac{\sum_{i=1}^N (x_i - \mu)^3}{N(\sigma)^3}$ (2.29)
<i>kurtosis2</i>	$\mu$ : örüntünün ortalama değeri <i>N</i> : örüntüdeki veri sayısı $x_i$ : örüntüdeki <i>i</i> . Veri $\sigma$ : örüntünün standart sapması	$kurtosis2 = \frac{\sum_{i=1}^N (x_i - \mu)^4}{N(\sigma)^4}$ (2.30)

Çizelge 2.3. CCP tanıma için kullanılan bazı özneliklerin adı, formül açıklaması ve formülü (devam)

Örüntüyü 4 gruba ayırarak elde edilen öznelikler		
<i>ASL</i>	Orta nokta çiftlerinin altı kombinasyonundan geçen düz çizgilerin eğimlerinin ortalaması $s_{jk}$ : eğim	$ASL = \frac{1}{6} \sum_{j < k} s_{jk}$ ( $j = 1,2,3; k = 2,3,4$ ) (2.31)
<i>AASL</i>	Orta nokta çiftlerinin altı kombinasyonundan geçen düz çizgilerin eğimlerinin ortalamasının mutlak değeri $s_{jk}$ : eğim	$AASL = \left  \sum_{j < k} \frac{s_{jk}}{6} \right $ ( $j = 1,2,3; k = 2,3,4$ ) (2.32)
<i>SRANGE</i>	Altı çiftli orta nokta kombinasyonundan geçen düz çizgilerin en büyük ile en küçük eğim değeri farkı $s_{jk}$ : eğim	$SRANGE = \max(s_{jk}) - \min(s_{jk})$ ( $j = 1,2,3; k = 2,3,4; j < k$ ) (2.33)
<i>ABL</i>	Altı gözlem alt kümesine ait LS çizgilerinin ortalama eğimi $b_{jk}$ : LS çizgisi eğimi	$ABL = \frac{1}{6} \sum_{j < k} b_{jk}$ ( $j = 1,2,3; k = 2,3,4$ ) (2.34)
<i>BRANGE</i>	Altı gözlem alt kümesine uyan LS çizgilerinin en büyük ile en küçük eğim farkı $b_{jk}$ : LS çizgisi eğimi	$BRANGE = \max(b_{jk}) - \min(b_{jk})$ ( $j = 1,2,3; k = 2,3,4; j < k$ ) (2.35)
<i>REAE</i>	Bölgelere ayrılmamış gözlemlere ait LS çizgisinin hata karelerinin ortalama toplamı (Mean Sum of Squares of Errors, MSE) ile altı gözlem alt kümesine ait LS çizgilerinin ortalama MSE'si oranı	$REAE = \frac{MSE}{\frac{1}{6} \sum_{j < k} MSE_{jk}}$ ( $j = 1,2,3; k = 2,3,4$ ) (2.36)
<i>RVAE</i>	Gözlemlerin varyansının altı gözlem alt kümesine ait LS çizgilerinin MSE'lerinin ortalamasına oranı	$RVAE = \frac{SD^2}{\frac{1}{6} \sum_{j < k} MSE_{jk}}$ ( $j = 1,2,3; k = 2,3,4$ ) (2.37)
<i>DABL</i>	Dört gözlem alt kümesine ait LS çizgilerinin ortalama eğimi $db_j$ : LS çizgisi eğimi	$DABL = \frac{1}{4} \sum_{j=1}^4 db_j$ ( $j = 1,2,3,4$ ) (2.38)

Çizelge 2.3. CCP tanıma için kullanılan bazı özniteliklerin adı, formül açıklaması ve formülü (devam)

<i>DBRANGE</i>	Dört alt gözlem kümesine ait LS çizgilerinin en büyük ile en küçük eğim farkı <i>db<sub>j</sub></i> : LS çizgisi eğimi	$DBRANGE = \max(db_j) - \min(db_j)$ ( $j = 1,2,3,4$ ) (2.39)
<b>Örüntüyü 2 gruba ayırarak elde edilen öznitelikler</b>		
<i>AABPE</i>	Çizgi parçalarının mutlak ortalama eğimi <i>B<sub>j</sub></i> : LS çizgisi eğimi	$AABPE = \left  \frac{1}{2} \sum_{j=1}^2 B_j \right $ ( $j = 1,2$ ) (2.40)
<i>ABDPE</i>	Örüntüye ait LS çizgisi ile iki pencere içindeki örüntülere ait çizgi parçaları arasındaki mutlak eğim farkı <i>B<sub>j</sub></i> : LS çizgisi eğimi <i>B</i> : örüntü LS çizgisi eğimi	$ABDPE = \left  B - \frac{1}{2} \sum_{j=1}^2 B_j \right $ ( $j = 1,2$ ) (2.41)
<i>SASDPE</i>	Örüntüye ait LS çizgisi ile ayrı çizgi parçaları arasındaki mutlak eğim farkının toplamı <i>B</i> : örüntü LS çizgisi eğimi <i>B<sub>j</sub></i> : LS çizgisi eğimi	$SASDPE = \sum_{j=1}^2  B - B_j $ ( $j = 1,2$ ) (2.42)
<i>SASPE</i>	İki çizgi parçasının mutlak eğimlerinin toplamı <i>B<sub>j</sub></i> : LS çizgisi eğimi	$SASPE = \sum_{j=1}^2  B_j $ ( $j = 1,2$ ) (2.43)
<i>REPEPE</i>	Örüntüye ait LS çizgisinin MSE oranı ile iki pencereye ait LS çizgilerinin toplanmış MSE (PMSE)'sine oranı	$REPEPE = \frac{MSE}{PMSE}$ (2.44)
<i>RVPEPE</i>	İki pencereye ait LS çizgilerinin PMSE'si ile örüntü varyans oranı	$RVPEPE = \frac{SD^2}{PMSE}$ (2.45)

Yukarıdaki açıklamalarda ve Çizelge 2.3'te verilen öznitelikler literatürde çalışmalarda kullanılmıştır. Pham ve Wani (1997), eğim, *nc2*, *nlc*, *cmember*, *a<sub>pm</sub>*, *a<sub>ps</sub>*, *a<sub>ss</sub>*, *a<sub>s</sub>* ve *sd* özniteliklerini kullanmışlardır. Wani ve Rashid (2005), eğim, *nc2*, *nlc*, *cmember*, *a<sub>pm</sub>*, *a<sub>ps</sub>*, *a<sub>ss</sub>*, *a<sub>s</sub>* ve *sd* özniteliklerini kullanmışlardır. Gauri ve Chakraborty (2007), eğim, *RVE*, *ACLPI*, *ACLMLC*, *ALSPI*, *ALSLSC*, *RACLALS*, *PMLC*, *PLSC*, *PSMLSC*, *RDIST*, *AASBP*,

*ASL, SRANGE, ABL, BRANGE, REAE, RVAE, DABL, DBRANGE, AABPE, ABDPE, SASDPE, SASPE, REPEPE* ve *RVPEPE* özniteliklerini kullanmışlardır. Ebrahimzad ve Ranaee (2009), eğitim, *PMLC, PLSC, skew1* ve *kurtosis1* özniteliklerini kullanmışlardır. Wang vd. (2009), eğitim, *PMLC, PLSC, skew2* ve *kurtosis2* özniteliklerini kullanmışlardır. Gauri (2010), eğitim, *ACLPI, PSMLSC* ve *AASL* özniteliklerini kullanmışlardır.

Öznitelik-tabanlı yöntemin ikinci aşamasında, elde edilen öznitelikler kullanılarak örüntü tanıma gerçekleştirilir. Pham ve Wani (1997) ikinci aşama için sezgisel, tümevarım ve sinir ağı tekniklerini, Gauri (2010) ağ tabanlı vektör niceleme öğrenmeyi, Wani ve Rashid (2005) dağıtılmış ve sinerjik sinir ağı yapısına dayanan paralel bir algoritmayı örüntü tanıma için kullanmıştır.

Bilgi tabanlı sistemlerde eğitim verisi ile karar ağacı öğrenme yaklaşımının kullanıldığı başarılı çalışmalar bulunmaktadır (Zorriassatine vd., 2005; Wang vd., 2008; Wang vd., 2009; Hachicha ve Ghorbel'dan (2012); Guh vd., 2005; Guh, 2005). Öznitelik-tabanlı yönteminin ikinci aşamasında da karar ağacı öğrenme yaklaşımları örüntü tanımanın gerçekleştirilmesinde kullanılmaktadır (Gauri ve Chakraborty, 2007; Wang vd., 2009).

### 3. MATERYAL VE YÖNTEM

Bu bölüm altında ilk olarak OTA için veri odaklı anomali tespiti kapsamında kullanılan Akıllı fabrika ve Robotik Laboratuvarı, oluşturulan Gazebo benzetim ortamı, oluşan veri içeriği ve kullanılan teknolojiler hakkında bilgi verilmektedir. Devam eden bölümde ise fabrika seviyesinde OTA'ların oluşturduğu trafikte ve OTA seviyesinde konumlandırma sistemlerinde veri odaklı anomali tespiti için önerilen yöntemler verilmektedir.

#### 3.1. Materyal

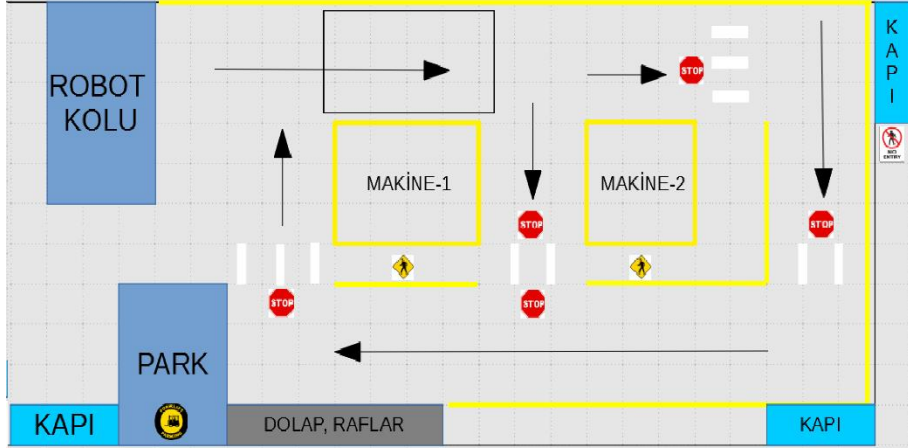
Bu bölümde Gazebo benzetim ortamı ve akıllı fabrika laboratuvarı ile oluşturulan veri kümeleri ve akıllı fabrikalardaki OTA'lar için önerilen veri odaklı anomali tespiti yöntemlerinde kullanılacak materyaller ile ilgili açıklamalar verilmektedir.

##### 3.1.1. Laboratuvar ortamı

Eskişehir Osmangazi Üniversitesi'nde Endüstri 4.0 konularında araştırma yapmak üzere Türkiye Bilimsel ve Teknolojik Araştırma Kurumu'nun (TUBİTAK) 116E731 nolu "Akıllı fabrikalar için otonom taşıyıcılar ve gerekli insan-makine ve makine-makine arayüzlerinin geliştirilmesi" projesi kapsamında Akıllı Fabrika ve Robotik Laboratuvarı (IFARLAB) (ESOGÜ, 2020) kurulmuştur. Şekil 3.1 (a)'da IFARLAB'da bulunan akıllı fabrika ortamı ve OTA, Şekil 3.1 (b)'de ise IFARLAB ortam yol grafiği verilmektedir. OTA ile ilgili laboratuvar ortamı testleri burada gerçekleştirilmiştir.



a)



b)

Şekil 3.1. (a) IFARLAB Akıllı fabrika ortamı ve OTA, (b) IFARLAB ortam ve yol şeması

OTA, IFARLAB’da taşıma görevlerini yerine getirirken ortamda bulunan belirli yolları kullanmaktadır (Bkz. Şekil 3.1 (b)). OTA seviyesinde konumlandırma sistemlerinde veri odaklı anomali tespiti için kullanılacak veri kümesi de OTA’nın fabrika içerisinde hareketi sırasında kullandığı yollardan biri seçilerek oluşturulmuştur. Veri kümesi, Şekil 3.1 (b)’de siyah dikdörtgen ile belirtilen alanda bulunan yol için OTA’nın yolu kullanım süresince konumlandırma sistemlerinden elde edilen enlem, boylam ve zaman bilgisinden oluşmaktadır. Verilerin elde edilmesi için OTA IFARLAB akıllı fabrika ortamında gezdirilerek üç farklı konumlandırma sisteminden elde edilen enlem, boylam ve zaman bilgisi kaydedilerek veri kümesi oluşturulmaktadır. Toplamda 24.357 adet veri elde

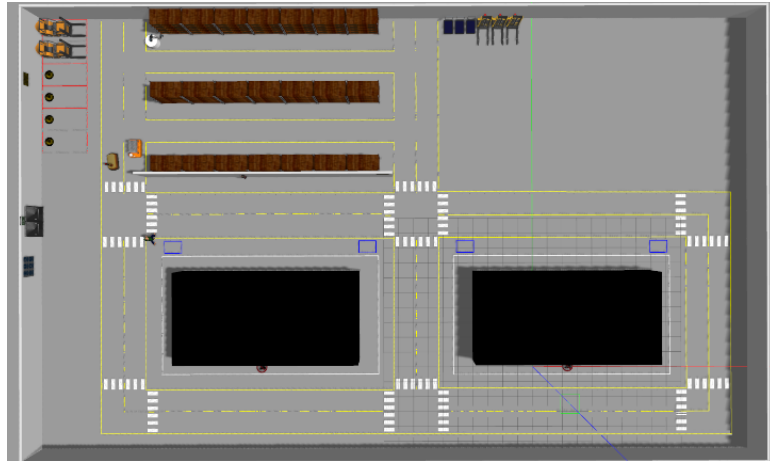
edilmektedir. Veri kümesinde yer alan konumlandırma sistemleri ve açıklamaları şu şekildedir:

- Odometre: Zaman içindeki konum değişikliğini tahmin etmek için hareket sensörlerinden gelen veriler,
- AMCL: İki boyutlu hareketli bir robot için olasılıksal bir yerleştirme sisteminden gelen veriler,
- UWB: Ultra Geniş Bant konumlandırmadan gelen verilerdir.

Oluşturulan veri kümesi, OTA seviyesinde konum sistemlerinde zaman içinde olan anormal değişimlerin veri odaklı anomali tespiti için önerilen yöntemde kullanılmaktadır. Yöntemin uygulanmasında “Python” programlama dili kullanılmıştır. Python, UCB algoritması, İPK yaklaşımlarının geliştirilmesinde ve sonuçların gösterilmesinde kullanılmıştır.

### 3.1.2. Benzetim ortamı

Fabrika seviyesinde akıllı fabrikada OTA’ların oluşturduğu trafikte oluşabilecek anormal durumların tespiti kapsamında kullanılacak normal ve anormal durumların bilindiği yani etiketli veri kümesi oluşturulmuştur. Veri kümesinin üretilmesinde IFARLAB ile bağlantılı Gazebo ortamında oluşturulmuş akıllı fabrika benzetim ortamı baz alınmıştır. Gazebo ortamında oluşturulmuş akıllı fabrika benzetim ortamına ait görsel Şekil 3.2 verilmektedir.



Şekil 3.2. GAZEBO ortamında oluşturulmuş akıllı fabrika sanal ortamı



Şekil 3.2’de verilen benzetim ortamından bir adet yaya geçidi bulunan yol ve bir adet doldurma boşaltma noktası bulunan yol seçilmiş ve bu yollar için on adet öznitelik içeren farklı senaryolara göre veri kümesi MATLAB kullanılarak oluşturulmuştur. Veri kümesindeki öznitelikler; zaman aralığı, ortalama hız, yol tipi, yol id, yol içeriği, aktiflik durumu, hız limiti, yol durumu, yaya yoğunluğu ve işlem süresidir. Özniteliklerin içeriği ve yapısı ile ilgili bilgiler Çizelge 3.1’de verilmektedir. Daha sonra farklı senaryolara göre üretilen veriler bilinmeyen anomali (araç kaynaklı), yavaşlama anomalisi (yaya yoğunluğu kaynaklı), yavaşlama anomalisi (araç kaynaklı), durma anomalisi (doldurma boşaltma kaynaklı) ve yok olarak etiketlenmiştir. Toplamda 2506 adet veri üretilmiştir.

Çizelge 3.1. Veri kümesine ait öznitelikler

Öznitelik Adı	İçeriği	Açıklama
Zaman aralığı	(08:00 – 16:00)	Verinin zamanını belirtir. 15’şer dakikalık aralıklar ile 16:00’a kadar değer alabilir. (HH:MM)
Ortalama hız	(0-20)	Yola ait ortalama hız değerini belirtir. Km/sa
Yol tipi	(0/1)	Yolda bulunan şerit sayısı (0 -Tek Şerit,1- Çift Şerit)
Yol id	(I/II)	Yola ait id değeridir.
Yol içeriği	(0/1)	Yolun içeriğine özgü 0 veya 1 değerini alır (0: yaya yolu, 1: doldurma boşaltma noktası).
Aktiflik durumu	(0/1)	Yolda bulunan doldurma boşaltma noktasının aktif olup olmama durumunu belirtir (0: pasif, 1: aktif).
Hız limiti	(0-20)	Yola ait hız limit değeridir.
Yol durumu	(0/1)	Yolun açık veya kapalı olma durumunu belirtir (0: kapalı, 1: açık).
Yaya yoğunluğu	(0-50)	Yolun yaya yoğunluğu değeridir.
İşlem süresi	(0-15)	Yolda bulunana doldurma boşaltma noktasının aktif olduğu süreyi belirtir. Birimi dakikadır.
Anomali Nedeni	Yok Araç Araç Yavaşlama Yaya Yoğunluğu Doldur Boşalt	Tespit edilecek anomali tipleri

Oluşturulan veri kümesi fabrika seviyesinde OTA'ların oluşturduğu trafikte anormal durumların tespiti için önerilen veri odaklı anomali tespiti yöntemlerinde kullanılmıştır. Bu yöntemler karar ağacı ve bulanık mantık tabanlı yöntemlerdir. Yöntemlerde kullanılan teknolojiler ile ilgili bilgiler Çizelge 3.2'de verilmektedir.

Çizelge 3.2. Veri kümesinin kullanıldığı yöntem ve teknolojiler

Yöntem	Kullanılan Teknolojiler ve Açıklaması
Bulanık Mantık Tabanlı Anomali Tespiti	<b>Python programlama dili:</b> Bulanık Mantık algoritmasının uygulanması ve sonuçların gösterilmesinde kullanılmıştır. <b>IBM SPSS Statistics 22:</b> Alıcı Operatör Karakteristikleri Eğrisi (Receiver Operator Characteristics Curve, ROC) analizinde kullanılmıştır.
Karar Ağacı Algoritması ile Anomali Tespiti	<b>WEKA:</b> Karar ağacı algoritmasının eğitimi ve test için kullanılmıştır. <b>C# programlama dili:</b> WEKA'nın C# uygulamasına entegre edilmesi ve sonuçların gösterilmesinde kullanılmıştır.

### 3.2. Akıllı Fabrikalarda OTA'ların Oluşturduğu Trafikteki Anormal Durumların Tespiti

Bu bölümde akıllı fabrikalardaki OTA'ların fabrika seviyesinde oluşturduğu trafikteki anormal durumların tespiti için denetimli öğrenme tabanlı yaklaşımlar verilmektedir. OTA'ların fabrika içerisinde kullandığı yollarda meydana gelebilecek; anormal yaya yoğunluğu, doldurma boşaltma noktalarında anormal bekleme, OTA kaynaklı anormal yavaşlama ve OTA kaynaklı nedeni bilinmeyen anormal durumların tespiti için karar ağacı, anomali “var” veya “yok” olarak durum tespiti için ise bulanık mantık tabanlı anomali tespit yöntemi önerilmektedir. Yöntemler ile ilgili detaylı bilgi devam eden bölümlerde verilmektedir.

#### 3.2.1. Karar ağacı ile anomali tespiti

Endüstri 4.0 ile gelişmekte olan akıllı fabrikaların en önemli bileşenlerinden biri OTA'dır. OTA'nın fabrika içerisinde taşıma görevlerini gerçekleştirirken karşılaşılabileceği bekleme ve durma gibi anormal durumların tespit edilmesi sistem için önemlidir. Çünkü bu

araçlar genelde insan gözetiminden bağımsız büyük fabrika ortamlarında çalışmaktadırlar. Bu nedenle OTA'ların oluşturduğu trafikte anomali tespiti çalışılmıştır. Çalışmasını fabrika ortamında gerçekleştiren OTA için ortam kaynaklı oluşabilecek anormal durumlar şu şekilde örneklenebilir:

- Fabrika içerisinde bulunan yaya geçit noktalarında oluşabilecek beklenmedik yoğunluk
- Fabrika ortamında oluşabilecek beklenmedik araç yoğunluğu
- Fabrika ortamında bulunan doldurma boşaltma noktalarında oluşabilecek arızalar
- Fabrika ortamında OTA'ın kullandığı yollar üzerinde olmaması gereken engeller
- Fabrika ortamına ait ısı, doku vb. gibi durumların OTA çalışmasını kötü yönde etkilemesi (örnek olarak sensörlerin çalışmasını olumsuz etkileyen koşullar verilebilir)

OTA kaynaklı durumlar ise şu şekilde örneklenebilir:

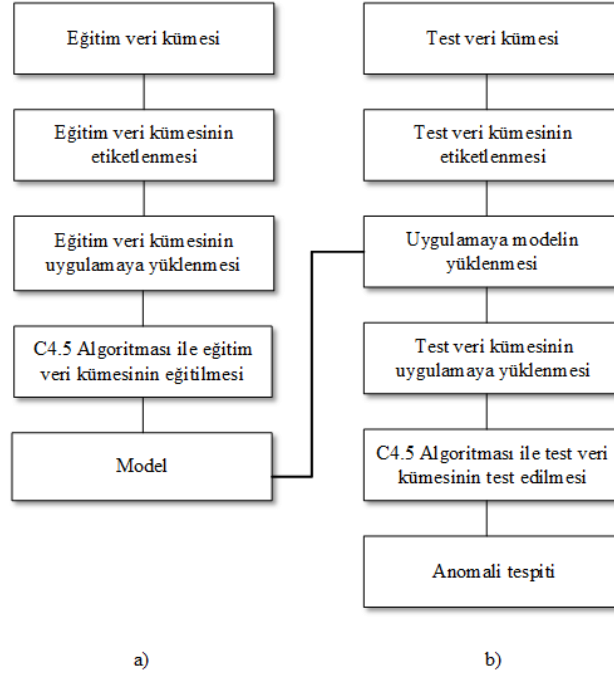
- OTA mekanik bozulma (örnek sensör arızaları)
- OTA yazılım sorunları (örnek sensör değerlerinin yanlış alınması)
- OTA'a dışarıdan bir saldırıda bulunulması verilebilir (örnek sensör değerlerinin değiştirilmesi).

Yukarıda verilen durumların oluşması durumunda sistemin olumsuz etkilenmesini önlemek için denetimli öğrenme kapsamından sınıflandırma tabanlı anomali tespiti önerilmiştir. Tespit edilecek OTA'ın fabrika içerisinde karşılaşılabileceği anormal durumlar:

- Yaya geçitlerinde oluşan beklenmeyen yoğunluk
- Doldurma boşaltma noktalarında beklenmeyen durma
- OTA kaynaklı yavaşlama veya
- OTA kaynaklı nedeni tam bilinmeyen durumlar olarak belirlenmiştir.

Bu amaç için uygun oluşturulan veri kümesi (Bkz. Bölüm 3.1.2) kullanılarak anormal durumların tespiti için karar ağacı algoritması kullanılmıştır (Örnek vd., 2018).

Karar ağacı algoritması olan C4.5 ile sınıflandırma tabanlı anomali tespiti iki aşamada gerçekleşmektedir. Bu aşamalar eğitim ve testir. Şekil 3.3'te eğitim ve test aşaması verilmektedir.



Şekil 3.3. Anomali tespiti akış diyagramı: (a) sistem eğitimi, (b) test verisi üzerinden anomali tespiti

Şekil 3.3 (a)'da verilen eğitim sürecinde:

- İlk olarak etiketlenmiş veri kümesi eğitim ve test veri kümesi olarak ikiye ayrılır.
- Eğitim veri kümesi yüklenir.
- Veri kümesi C4.5 algoritması kullanılarak eğitilir.
- Eğitim sonucunda model elde edilir.
- Şekil 3.3 (b)'de verilen test aşamasına geçilir.

Şekil 3.3 (b)'de verilen eğitim sürecinde:

- Etiketli test veri kümesi ve oluşturulan model uygulamaya yüklenir.
- Model kullanılarak test veri kümesindeki veriler için sınıf tahmini, anomali tespiti, gerçekleştirilir.

### 3.2.2. Bulanık mantık tabanlı anomali tespiti

Akıllı fabrikalarda OTA'ların oluşturduğu trafikte anormal durumların tespiti için önerilen ikinci yöntem bulanık mantık tabanlı anomali tespittir (Örnek vd., 2020). Bu

yöntemde ilk olarak bulanık mantık kullanılarak anomali olma oranı bulunur. Daha sonra anomali oranı analiz edilerek anomali tespiti gerçekleştirilmektedir.

Bulanık mantık ile anomali tespitinde benzetim ortamında oluşturulan veri kümesi kullanılmaktadır (Bkz. Bölüm 3.1.2). Ancak veri kümesinin tüm öznitelikleri kullanılmamıştır. Kullanılan öznitelikler; zaman ortalama hız, işlem süresi ve yaya yoğunluğudur. Bulanık mantık dört aşamadan oluşmaktadır. Bunlar:

- Bulanıklaştırma (Fuzzification)
- Kurallar (Rules)
- Çıkarım (Inference)
- Berraklaştırma (Defuzzification)

Aşamalarda yer alan bulanık mantık unsurları da şu şekildedir (Zadeh, 1988):

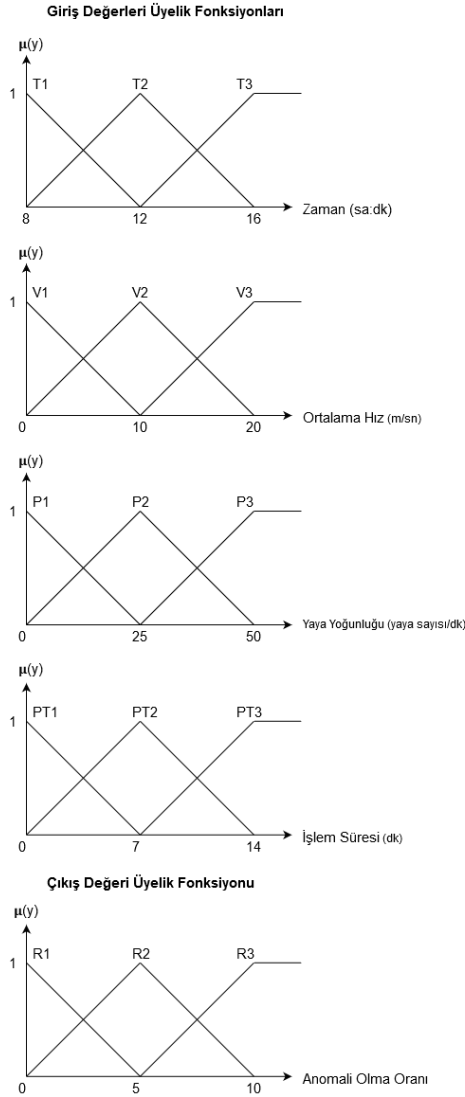
- Kesin giriş verileri,
- Dilsel değişkenler ve terimler,
- Bulanık kurallar,
- Üyelik fonksiyonları,
- Kesin çıktı verileri.

Bulanık Mantık tabanlı anomali tespitinde kullanılacak bulanık mantık unsurları kesin giriş verileri, dilsel değişkenler, bulanık kurallar, üyelik fonksiyonları ve kesin çıktı içerikleri ise şu şekildedir (Örnek vd., 2020)

- Kesin giriş verileri: Zaman, ortalama hız, işlem süresi ve yaya yoğunluğu
- Kesin çıkış verileri: Anomali durumu (“var” veya “yok”) ve anomali oranı
- Giriş verileri dilsel değişkenler: Zaman için  $T1$ ,  $T2$  ve  $T3$ , ortalama hız için  $V1$ ,  $V2$  ve  $V3$ , işlem süresi için  $PT1$ ,  $PT2$  ve  $PT3$ , yaya yoğunluğu için  $P1$ ,  $P2$  ve  $P3$
- Çıkış verisi dilsel değişkenler: Anomali durumu (“var”, “yok”) ve anomali oranı için  $R1$ ,  $R2$  ve  $R3$
- Bulanık kurallar: Oluşturulan bulanık kurallardan örnekler aşağıda verilmektedir (Örnek vd.’den, 2020).

Eğer	T1	ve	V1	ve	P1	ve	PT1	ise	R2
Eğer	T1	ve	V1	ve	P1	ve	PT2	ise	R3
Eğer	T1	ve	V1	ve	P1	ve	PT3	ise	R1
Eğer	T1	ve	V1	ve	P2	ve	PT1	ise	R2
Eğer	T1	ve	V1	ve	P2	ve	PT2	ise	R1
Eğer	T2	ve	V1	ve	P1	ve	PT1	ise	R2
Eğer	T2	ve	V1	ve	P1	ve	PT2	ise	R2
Eğer	T2	ve	V1	ve	P1	ve	PT3	ise	R3
Eğer	T2	ve	V1	ve	P2	ve	PT1	ise	R3
Eğer	T2	ve	V1	ve	P2	ve	PT2	ise	R2
Eğer	T3	ve	V1	ve	P1	ve	PT1	ise	R2
Eğer	T3	ve	V1	ve	P1	ve	PT2	ise	R1
Eğer	T3	ve	V1	ve	P1	ve	PT3	ise	R2
Eğer	T3	ve	V1	ve	P2	ve	PT1	ise	R2
Eğer	T3	ve	V1	ve	P2	ve	PT2	ise	R1

- Üyelik fonksiyonları: Girişler ve çıkış için oluşturulan üyelik fonksiyonları Şekil 3.4'te verilmektedir.



Şekil 3.4. Giriş değerleri ve çıkış değeri üyelik fonksiyonları (Örnek vd.'den, 2020)

Tüm gerekli tanımlamaların yapılmasının ardından bulanık mantık aşamalarında belirlenen unsurlar kullanılarak anomali tespiti gerçekleştirilir. Bulanıklaştırma, kural değerlendirilmesi, kural çıktılarının toplanması ve berraklaştırma aşamaları “Python” programlama dili kullanılarak yapılmıştır:

- Bulanıklaştırma aşaması: kesin giriş verileri üyelik fonksiyonlarında kullanılarak her bir giriş verisinin belirlenen dilsel değişkenler için üyelik dereceleri, bulanıklaştırılmış girdiler, belirlenir.
- Kural değerlendirmesi aşaması: bulanıklaştırılmış giriş verilerinin oluşturulan kurallarda belirlenen dilsel çıkış verileri elde edilir.
- Kural çıktılarının toplanması aşaması: tüm kural çıktıları ve bulanıklaştırılmış çıkış verileri Max-dot yöntemi ile birleştirilir (Mendel, 1995).
- Berraklaştırma aşaması: ağırlık merkezi yöntemi kullanılarak bulanıklaştırılmış çıkış verilerinden tek bir değer elde edilir (Mendel, 1995). Bu değer anomali oranını belirtir.

Elde edilen anomali oranı ile verilerin anomali durumu (“var” veya “yok”) “IBM SPSS Statistics 22” uygulaması ile ROC analizinde kullanılır. ROC analizi ile anomali tespiti için uygun eşik değeri belirlenir. Daha sonra belirlenen eşik değeri ile anomali tespiti gerçekleştirilir (Örnek vd.’den, 2020).

### **3.3. OTA’lar için Konumlandırma Sistemindeki Anormal Durum Tespiti**

Endüstriyel sistemlerde endüstri 4.0 ile birlikte akıllı fabrika sistemlerinde otonom taşıyıcı robotların kullanımı artmaktadır. Bu bağlamda Akıllı fabrikalarda iç lojistikte malzeme taşıma, alma vb. görevlerde OTA’ların kullanımı artmaktadır. Robotik sistemlerde olduğu gibi OTA’larda görevlerini yerine getirirken konum bilgisi kritik öneme sahiptir. Hedefe güvenli ve doğru bir şekilde varabilmek için hareketleri boyunca doğru konum bilgisine sahip olmaları veya bunun doğrulanması gerekir. OTA’lar ise tüm işlevlerini kendi sistemleri ile kararlar alarak gerçekleştirdiği için bu durum daha da önem arz etmektedir. OTA’ların konumunu tespit edebilmek için kullandığı farklı konumlandırma sistemlerinde zaman içerisinde meydana gelecek herhangi bir değişimin anormal bir durum olarak tespit edilmesi önemlidir. Çünkü fabrika içi fiziksel ortam

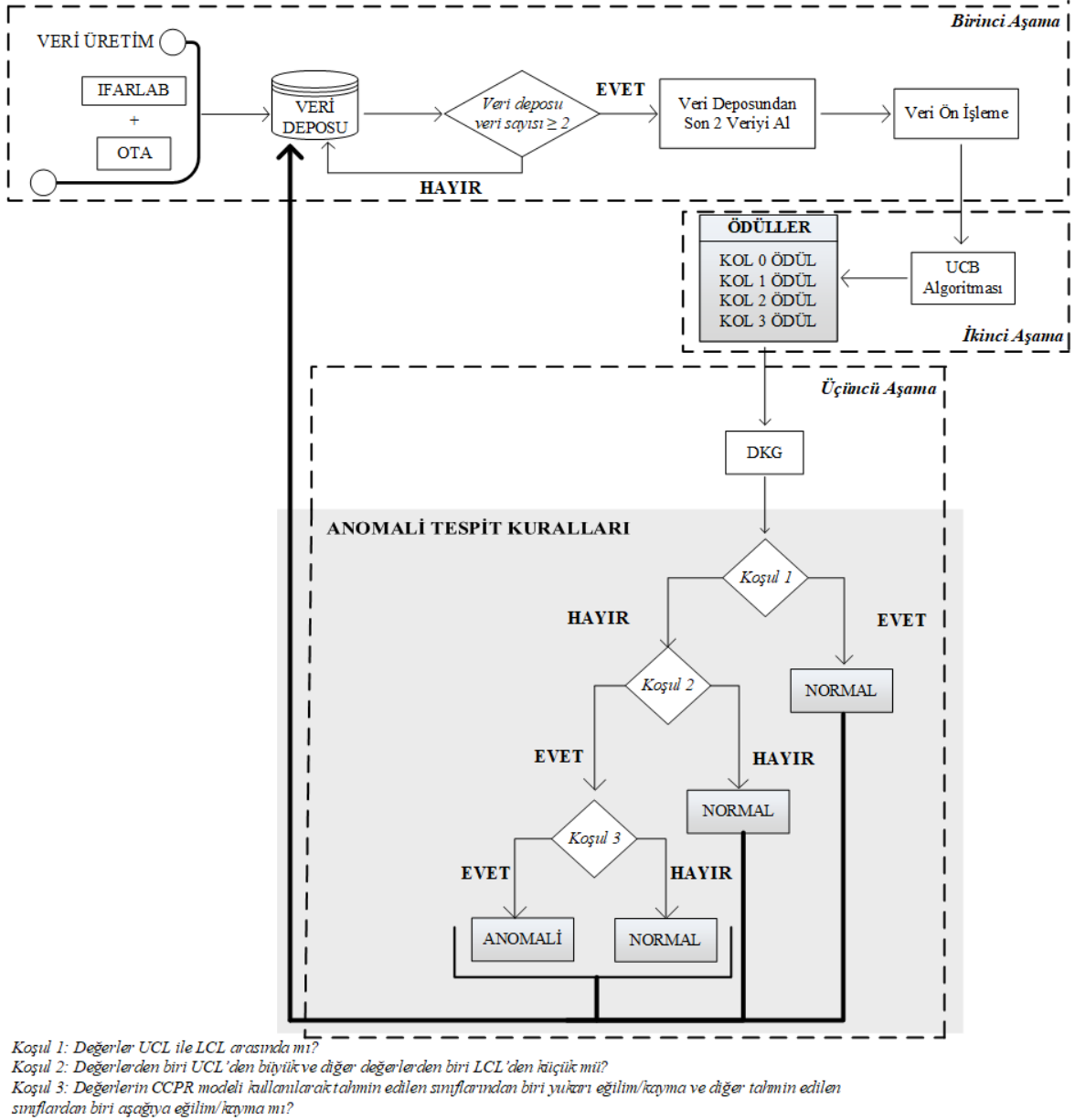
değişimlerine bağlı olarak RF tabanlı konumlandırma sistemlerinde zaman içinde hassasiyet değişimi olabilmektedir. Bu nedenle birden fazla konumlandırma yaklaşımından gelen konum bilgisi değerlendirilerek kullanılmaktadır. Buradan yola çıkılarak bu çalışmada, OTA için güvenilir konumlandırma sistemlerinin belirlenmesi ve bu durumun zaman içerisinde meydana gelecek herhangi bir değişiklik nedeniyle değişiminin tespiti için dinamik olarak durumları öğrenebilecek değişimleri fark edebilmemizi ve yeni bir güvenilir konumlandırma sisteminin gelmesi durumunda bu durumu öğrenebilecek bir sistem geliştirilmesi hedeflenmektedir. Bu amaçla pekiştirmeli öğrenme yöntemlerinden ÇKS altında yer alan UCB algoritması kullanılmıştır.

OTA'nın güvendiği konumlandırma sistemlerinin karakteristiğini öğrenebilmek için IFARLAB akıllı fabrika test ortamından elde edilen konum verileri kullanılmıştır (Bkz. Bölüm 3.1.1). Bu veri kümesinde laboratuvarında bulunan akıllı fabrika test ortamında yer alan OTA'nın ortamdaki hareketleri sonucunda üç farklı konumlandırma sisteminden (UWB, AMCL ve Odometre) elde edilen veriler bulunmaktadır. Her konumlandırma sistemi için her bir zaman için sayısal enlem ve boylam değerleri bulunmaktadır. Üç farklı konumlandırma sistemi baz alınarak güvenilir konumlandırma sistemleri alt kümeleri şu şekilde belirlenmiştir:

- V3: UWB konumlandırma sistemine güvenilmediği durum
- V2: AMCL konumlandırma sistemine güvenilmediği durum
- V1: Odometre konumlandırma sistemine güvenilmediği durum
- V0: Tüm konumlandırma sistemlerine güvenildiği durum

Şekil 3.5'te OTA için en güvenilir konumlandırma sistemlerini öğrenme ve güvenilen konumlandırma sistemlerinin değişmeye başlamasının anormal durum olarak tespit edilmesi için veri odaklı pekiştirmeli öğrenme tabanlı önerilen yönteme ait akış şeması verilmektedir.





Şekil 3.5. Pekiştirmeli öğrenme tabanlı veri odaklı önerilen anomali tespiti akış şeması

Şekil 3.5'te verilen ilk aşamada verinin üretilmesi ve veri işleme işlemleri gerçekleştirilmektedir. OTA aldığı görevleri IFARLAB ortamında yerine getirirken belirlenen bir yol için OTA'nın üç farklı konumlandırma sisteminden elde edilen veriler veri tabanında depolanmaktadır. Veri tabanına her eklemede veri içeriğinde sırasıyla:

- Zaman,
- Odometre konumlandırmadan elde edilen enlem bilgisi,
- Odometre konumlandırmadan elde edilen boylam bilgisi,
- AMCL konumlandırmadan elde edilen enlem bilgisi,

- AMCL konumlandırılmadan elde edilen boylam bilgisi,
- UWB konumlandırılmadan elde edilen enlem bilgisi,
- UWB konumlandırılmadan elde edilen boylam bilgisi bulunmaktadır.

Veri tabanında bulunan veri miktarı (sıra sayısı) ikiyi geçtikten sonra önerilen sistem bekleme aşamasını bitirerek aktif hale gelir. Aktif duruma gelindiğinde konumlandırma sistemlerinden elde edilen veriler veri tabanından alınarak işlenmesi gerçekleştirilir. Veri işleme aşamaları şu şekildedir:

- Veri tabanında bulunan son iki sıra veri alınır.
- Alınan iki sıra veri içerisinde her bir konumlandırma sistemi için ikişer adet enlem ve boylam değeri bulunur.
- Alınan iki sıra veri kullanılarak her konumlandırma sistemi için vektör değerleri elde edilir.
- Elde edilen vektör değerleri kullanılarak konumlandırma sistemlerinin birbirileri arasında bulunan açı değerleri hesaplanır:
  - Odometre ile AMCL arasındaki açı
  - Odometre ile UWB arasındaki açı
  - AMCL ile UWB arasındaki açı
- Elde edilen vektör değerleri kullanılarak konumlandırma sistemlerinin büyüklük değerleri hesaplanır:
  - Odometre vektör büyüklüğü
  - AMCL vektör büyüklüğü
  - UWB vektör büyüklüğü
- Her konumlandırma sistemi için alınan iki sıra veri (enlem ve boylam değerleri) kullanılarak konumlandırma sistemlerinin birbirileri arasında bulunan “Mahalanobis Mesafesi (MM)” hesaplanır:
  - Odometre ile AMCL arasındaki MM
  - Odometre ile UWB arasındaki MM
  - AMCL ile UWB arasındaki MM
- Her konumlandırma sistemi için alınan iki sıra veri (enlem ve boylam değerleri) kullanılarak konumlandırma sistemlerinin birbirileri arasında bulunan “Hausdorff Mesafesi (HM)” hesaplanır:
  - Odometre ile AMCL arasındaki HM

- Odometre ile UWB arasındaki HM
- AMCL ile UWB arasındaki HM

Şekil 3.5'deki ikinci aşamada Algoritma 1'de verilen UCB algoritması uygulanmaktadır. UCB algoritması temel olarak algoritmada tanımlanan kollardan en yüksek UCB değerini veren kolun seçilmesidir. Seçilen kol eğer istenilen durum ise ödül verilir, değilse ödül alamaz. Ödül alma ve seçilme durumu bir dahaki kolun seçiminde rol oynamaktadır. Çalışmada UCB algoritmasına kollar olarak belirlenen konumlandırma sistemleri alt kümeleri kullanılmaktadır. Bunlar:

- $V3$  (UWB konumlandırma sistemine güvenilmediği durum):  $KOL3$
- $V2$  (AMCL konumlandırma sistemine güvenilmediği durum):  $KOL2$
- $V1$  (Odometre konumlandırma sistemine güvenilmediği durum):  $KOL1$
- $V0$  (Tüm konumlandırma sistemlerine güvenildiği durum):  $KOL0$

İstenilen kol seçimi yapılması durumunda alınacak ödül ise elde edilen konumlandırma sistemleri arasındaki açı, MM, HM ve konumlandırma sistemi vektör büyüklük değerleri kullanılarak en iyi uyumu sağlayan konumlandırma sistemleri alt kümesine göre belirlenmektedir. En iyi uyumu sağlayan konumlandırma sistemleri alt kümesi beş adımda belirlenmektedir. İlk adımda, konumlandırma sistemleri vektör büyüklük değerlerinin tümü kullanılarak varyans hesaplanır,  $KOL0$ 'ı belirtir.  $KOL1$ ,  $KOL2$  ve  $KOL3$  sırasıyla; Odometre hariç, AMCL hariç ve UWB hariç diğer vektör büyüklükleri kullanılarak hesaplanan varyans değerini belirtir. İkinci adımda, konumlandırma sistemleri vektörleri arasındaki açı değerlerinin tümü ile ortalama hesaplanır,  $KOL0$ 'ı belirtir.  $KOL1$ ,  $KOL2$  ve  $KOL3$  sırasıyla; AMCL ile UWB, Odometre ile UWB ve Odometre ile AMCL vektörleri arasındaki açı değerini belirtir. Üçüncü adımda, konumlandırma sistemleri arasındaki MM değerlerinin tümü kullanılarak ortalama hesaplanır,  $KOL0$ 'ı belirtir.  $KOL1$ ,  $KOL2$  ve  $KOL3$  sırasıyla; AMCL ile UWB, Odometre ile UWB ve Odometre ile AMCL vektörleri arasındaki MM değerini belirtir. Dördüncü adımda, konumlandırma sistemleri arasındaki HM değerlerinin tümü kullanılarak ortalama hesaplanır,  $KOL0$ 'ı belirtir.  $KOL1$ ,  $KOL2$  ve  $KOL3$  sırasıyla; AMCL ile UWB, Odometre ile UWB ve Odometre ile AMCL vektörleri arasındaki HM değerini belirtir. Beşinci adımda, her adımda  $KOL0$ ,  $KOL1$ ,  $KOL2$  ve  $KOL3$  için hesaplanan değerler küçükten büyüğe sıralanır. Sıralanan değerler için küçükten büyüğe doğru sıfırdan üçe kadar numara verilir. Her adımda alınan numara bir önceki

adımında alınan numaranın üstüne eklenir. Son olarak dördüncü adımdan sonra kol değerlerinin elde ettikleri toplam değerler küçükten büyüğe sıralanır. En küçük değere sahip kol değeri en uyumlu kol olarak seçilir.

UCB tarafından seçilen kol ile konumlandırma sistemleri arasındaki açığı, MM, HM ve konumlandırma sistemi vektör büyüklük değerleri kullanılarak elde edilen değerlerin analizi sonucu seçilen kol yani konumlandırma sistemleri alt kümesi aynı ise ödül alınır değilse ödül alınmaz. Zaman içerisinde alınan ödül değerleri ile UCB karakteristiği, güvenilir konumlandırma sistemleri alt kümesini, öğrenmeye başlar. Amaç güvenilir konumlandırma sistemleri alt kümesinin öğrenilen karakteristiğe uymamaya yani ödül değerinde değişikliğin başladığı zamanın anormal bir durum olarak tespit edilmesidir.

<b>Algoritma 1</b> UCB sözde kodu		
0:	Giriş:	veri: işlenmiş konumlandırma sistemi verileri
1:	Tanımlama:	$Q_t(a)$ : $t$ turuna kadar elde edilen $a$ koluna ait ortalama ödül
2:		$N_t(a)$ : $t$ turuna kadar $a$ kolunun seçilme sayısı
3:		$R_t(a)$ : $t$ turuna kadar elde edilen $a$ koluna ait toplam ödül
4:		$ave\_r_a(t)$ : her $t$ turunda $a$ koluna ait ortalama ödül
5:		$A_t$ : $t$ turunda seçilen kol
6:		$c = 2$ : keşif katsayısı
7:		$K = 4$ : kol sayısı
8:		$N$ : gözlem sayısı
9:	Fonksiyonlar:	$Angle\_Magnitude\_MD\_HD(veri[t])$ : $t$ turunda data içinde bulunan
10:		verileri kullanarak en iyi konumlandırma sistemleri alt kümesini (kol) seçer
11:	Çıkışlar:	$ave\_r_a$
12:		<i>for</i> $t = 1, 2, \dots, N$ <i>do</i>
13:		<i>if</i> $t \leq K$ <i>do</i>
14:		$N_t(a)_{a \in \{1,2,3,4\}} = N_t(a)_{a \in \{1,2,3,4\}} + 1$
15:		<i>else do</i>
16:		$A_t = \operatorname{argmax}_{a \in \{1,2,3,4\}} \left( Q_t(a) + c \sqrt{\frac{\ln(t)}{N_t(a)}} \right)$
17:		$N_t(A_t) = N_t(A_t) + 1$
18:		
19:		

20:		$select\_arm = Angle\_Magnitude\_MD\_HD(veri[t])$
21:		
22:		$if\ select\_arm = A_t\ do$
23:		$R_t(A_t) = R_t(A_t) + 1$
24:		$Q_t(a)_{a \in \{1,2,3,4\}} = \frac{R_t(a)}{N_t(a)}$
25:		$ave\_r_a(t)_{a \in \{1,2,3,4\}} = Q_t(a)$
26:		$end\ for$

Şekil 3.5'deki üçüncü aşamada UCB algoritması sonucunda elde edilen kollara ait her adımda hesaplanan ortalama ödül değerleri İPK yaklaşımlarından Algoritma 2'de verilen Dinamik Kontrol Grafiği (DKG) ile kullanılarak kontrol grafiği limitleri ve grafiğe çizdirilecek değerler belirlenir.

Algoritma 2 DKG sözde kodu		
0:	Giriş:	$ave\_r_a_{a \in \{1,2,3,4\}}$ : her $t$ turunda $a$ koluna ait ortalama ödül
1:	Tanımlama:	$interval = 50$ : CL güncellenmesinde kullanılacak grup sayısı
2:		$m = 10$ : her grupta bulunan eleman, veri sayısı
3:		$N$ : gözlem sayısı
4:		$counter = 0$ : güncelleme zamanını saymak için (değişken)
5:	Fonksiyonlar:	$data\_grouping(data, m)$ : $data$ dizisindeki verileri $m$ 'lik gruplara ayırır
6:		$get\_data\_in\_last\_interval(ave\_r_a, interval * m)$ : $ave\_r_a_{a \in \{1,2,3,4\}}$
7:		dizisinde bulunan son $interval * m$ kadar veriyi bulur
8:		$mean(data)$ : $data$ içerisindeki her grubun ortalamasını hesaplar
9:		$skew(data)$ : $data$ 'nın çarpıklık (skewness) işaretini verir
10:		$kurtosis(data)$ : $data$ 'nın basıklık (kurtosis) değerini verir
11:		$std\_dev(data)$ : $data$ 'nın standart sapma değerini verir
12:		
13:		
14:	Çıktılar:	$data_{a \in \{1,2,3,4\}}$ : DKG grafiğinde yer alacak kollara ait veriler
15:		$ucl_{a \in \{1,2,3,4\}}$ : DKG grafiğinde her kol için ÜKL değeri
16:		$lcl_{a \in \{1,2,3,4\}}$ : DKG grafiğinde her kol için AKL değeri
12:		$for\ t = 1, 2, \dots, N\ do$
13:		$if\ t = 30\ do$

14:	$data_a = data\_grouping(ave\_r_a, m)$
15:	$data\_mean_{a \in \{1,2,3,4\}} = mean(data_a)$
16:	
17:	$centerline_{a \in \{1,2,3,4\}} = mean(data\_mean_a)$
18:	$skew\_sign_{a \in \{1,2,3,4\}} = skew(ave\_r_a)$
19:	$kurtosis\_value_{a \in \{1,2,3,4\}} = kurtosis(ave\_r_a)$
20:	
21:	$std\_dev\_value_{a \in \{1,2,3,4\}} = std\_dev(ave\_r_a)$
22:	
23:	$ucl_{a \in \{1,2,3,4\}} = centerline_a + ((3 + skew\_sign_a * (-1) *$
24:	$kurtosis\_value_a) * std\_dev\_value_a)$
25:	$lcl_{a \in \{1,2,3,4\}} = centerline_a - ((3 - skew\_sign_a * (-1) *$
26:	$kurtosis\_value_a) * std\_dev\_value_a)$
27:	
28:	<i>else if t &gt; 30 do</i>
29:	<i>if counter = m do</i>
30:	$data\_r_a = get\_data\_in\_last\_interval(ave\_r_a, interval * m)$
31:	$data_a = data\_grouping(data\_r_a, m)$
32:	$data\_mean_a = mean(data_a)$
33:	$centerline_a = mean(data\_mean_a)$
34:	$skew\_sign_a = skew(data\_r_a)$
35:	$kurtosis\_value_a = kurtosis(data\_r_a)$
36:	$std\_dev\_value_a = std\_dev(data\_r_a)$
37:	$data_a = mean(ave\_r_a(t))$
38:	$counter = counter + 1$
39:	<i>end for</i>
40:	
41:	

Çarpıklık ve basıklık değerlerinin hesaplanmasında kullanılan formüller şu şekildedir:

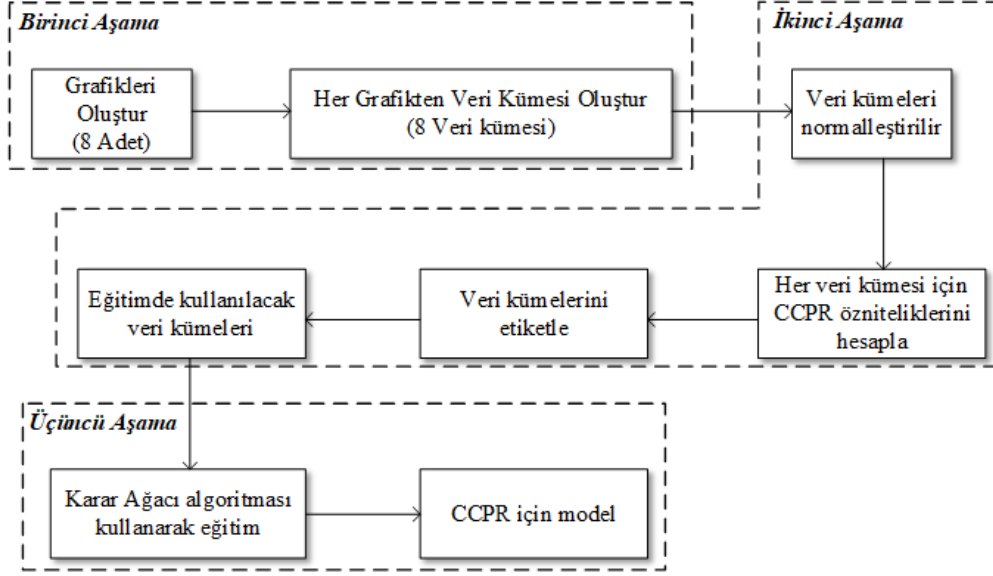
$$\text{Çarpıklık} = \frac{3(\text{Ortalama} - \text{Orta Değer})}{\text{Standart Sapma}}$$

(3.1)

$$Basıklık(x) = E \left[ \left( \frac{x - Ortalama}{Standart Sapma} \right)^4 \right] \quad (3.2)$$

Daha sonra oluşturulan anomali tespit kuralları kullanılarak sistem durumu “anomali” veya “normal” olarak tespit edilir. Anomali tespit kuralları üç koşul içermektedir. Birinci koşul, DKG’nde her kol ortalama ödül değeri için çizdirilen değerlerin hepsinin belirlenen kontrol limitleri arasında olup olmadığını kontrol eder. Eğer hepsi kontrol limitleri arasında ise durum normal kabul edilir ve birinci aşamaya dönülerek döngü devam eder. Ancak hepsi kontrol limitleri arasında değil ise ikinci koşula geçilir. İkinci koşul, DKG’nde her kol ortalama ödül değeri için çizdirilen değerlerden herhangi birisinin o kol için belirlenen ÜKL’nden yukarıda ve diğer değerlerden herhangi birisinin o kol için belirlenen AKL’nden aşağıda ise üçüncü koşula geçilir. Eğer değerler bu koşula uygun değil ise durum normal kabul edilir ve birinci aşamaya dönülerek döngü devam eder. Üçüncü koşulda ise ikinci aşamada kontrol limitleri dışında tespit edilen değerlerin örüntüsüne bakılmaktadır. Örüntü tespitinde Şekil 3.6’da akış şeması verilen CCPR modeli kullanılmaktadır. İkinci aşamada belirlenen kontrol limitleri dışına çıkan değerlerin örüntü tipleri CCPR modeli kullanılarak tahmin edilir. Eğer AKL’nden aşağıda olan değerlerin örüntü tipi aşağıya kayma veya aşağıya trend ve ÜKL’nden yukarıda olan değerlerin örüntü tipi yukarı kayma veya yukarı trend ise durum anomali olarak kabul edilir. Eğer değerlerin örüntü tipleri bu koşula uygun değil ise durum normal kabul edilir. Üçüncü koşul sonunda tekrar birinci aşamaya dönülerek döngü devam eder.

Şekil 3.6’da Şekil 3.5 üçüncü aşamasında kullanılan CCPR modelinin oluşturulmasında kullanılan yöntemin akış şeması verilmektedir.



Şekil 3.6. CCPR modelinin oluşturulması için akış şeması

Şekil 3.6’da verilen birinci aşamada farklı karakteristiklere sahip grafiklerin oluşturulması ve veri kümelerinin elde edilmesi gerçekleştirilmektedir. Dokuz çeşit grafik kullanılmaktadır:

- Normal
- Katmanlaşma
- Sistematik
- Döngüsel
- Sabit (y ekseninde sabit bir değer)
- Yukarı Trend
- Aşağı Trend
- Yukarı Kayma
- Aşağı Kayma

Oluşturulan dokuz çeşit grafikteki bilgiler kullanılarak her grafik çeşidi için 3000 adet veri içeren veri kümesi oluşturulur.

Şekil 3.6’da verilen ikinci aşamada veri kümelerinin işlenmesi gerçekleştirilmiştir. İlk olarak grafiklerden oluşturulan veri kümeleri normalize edilir. Elde edilen yeni veri kümeleri kullanılarak her grafik için örüntü tanıma parametreleri (öznitelikler) hesaplanır. Hesaplamanın yapılması için veri kümeleri her biri içerisinde on adet veri bulunduracak şekilde gruplara ayrılır. Parametrelerin hesaplanmasında ilk olarak dört grup veri ile



başlanır. Daha sonra her adımda bir ileri kaydırılarak yine dört grup veri alınarak hesaplamalara devam edilir. Bu durum dört grupluk kayan pencere (window) olarak açıklanabilir. Hesaplanan parametreler (öznitelikler) ve açıklamaları şu şekildedir:

- $a_{pm}$ : Dört gruba ait veriler ile Dört grubun ortalaması belirten çizgi ( $ml$ ) arasında kalan alan
- $a_{ps}$ : Dört gruba ait verileri kullanarak lineer regresyon yapılarak çizgi ( $pl$ ) ile 4 grubun ortalaması belirten çizgi ( $ml$ ) arasında kalan alan
- $a_{ss}$ : Dört gruba ait veriler  $pl$  arasında kalan alan
- $a_s$ : Dört grup için her gruba ayrı ayrı lineer regresyon yapılarak her grubun çizgisi ( $gl_i$   $i = \{1,2,3,4\}$ ) bulunur.  $gl_i$ 'ler kullanılarak her çizginin eğimi ( $gs_i$ ) bulunur. Tüm  $gs_i$ 'lerin ortalamasıdır.
- $sd$ :  $pl$  çizgisinin eğimi ile  $a_s$  değeri arasındaki farktır.
- $ACLPI$ :  $a_{pm}$  değeri dört grup içerisindeki toplam veri sayısının bir eksiğine bölünür. Elde edilen değer dört grupluk verinin standart sapmasının ( $std$ ) karesine bölünür.
- $MN-CROSS$ : Dört gruba ait tüm verilerin  $ml$  çizgisini kaç kere geçtiğini belirtir.
- $LS-CROSS$ : Dört gruba ait tüm verilerin  $pl$  çizgisini kaç kere geçtiğini belirtir.
- $PSMLSC$ :  $MN-CROSS$  ile  $LS-CROSS$  toplamının, dört grup içerisindeki toplam veri sayısının bir eksiğinin iki katına bölünmesi ile elde edilir.
- $LS-SLP$ : Dört grupluk verinin örüntüsünü temsil eden en küçük kareler çizgisinin eğimi
- $ACLMLC$ :  $a_{pm}$  değerinin  $MN-CROSS$  değerine bölümünden elde edilen değer  $std$  değerine bölümünden elde edilir.
- $ASL$ : Dört grup için her grubun orta noktası bulunur. Dört grubun kombinasyonu oluşturulur. Bunlar dört grup için şu şekildedir:
  - $\{1,2\}, \{1,3\}, \{1,4\}, \{2,3\}, \{2,4\}, \{3,4\}$
  - Kombinasyonların orta noktalarından geçen çizgi lineer regresyon ile bulunur. Çizginin eğimi hesaplanır. Bu işlem tüm kombinasyonlar için yapılır. Her kombinasyondan elde edilen eğim değerlerinin ortalaması alınır.
- $SRANGE$ :  $ASL$  hesaplamasında oluşturulan her kombinasyonun eğim değerlerinden en büyüğünün en küçüğünden çıkarılmasıyla elde edilir.
- $ABL$ : Dört grubun kombinasyonu oluşturulur. Bunlar dört grup için şu şekildedir:

- {1,2}, {1,3}, {1,4}, {2,3}, {2,4}, {3,4}
- Kombinasyon içindeki grup numarasına göre o gruplara ait verilerden geçen çizgi lineer regresyon ile bulunur. Çizginin eğimi hesaplanır. Bu işlem tüm kombinasyonlar için yapılır. Her kombinasyondan elde edilen eğim değerlerinin ortalaması alınır.
- *BRANGE*: *ABL* hesaplamasında oluşturulan her kombinasyonun eğim değerlerinden en büyüğünün en küçüğünden çıkarılmasıyla elde edilir.
- *ALSPI*:  $a_{ps}$  değerinin dört gruptaki toplam veri sayısının bir eksiğine bölümünden elde edilen değer *std* değerine bölünerek elde edilir.
- *AASBP*: Standart sapma açısından ardışık noktalardan geçen düz çizgilerin ortalama mutlak eğimi
- *REAE*: Lineer regresyon sonucu elde edilen *pl* için hata değerinin dört grup için oluşturulan kombinasyonlarında elde edilen çizgilere ait hata oranlarının ortalamasına bölümü ile elde edilir. Kombinasyonlar:
  - {1,2}, {1,3}, {1,4}, {2,3}, {2,4}, {3,4}
- *DABL*: Dört grup için dört farklı kombinasyon oluşturulur. Bunlar:
  - {1,2}, {1,2,3}, {3,4}, {2,3,4}
  - Her kombinasyona ait gruptaki veriler lineer regresyonda kullanılarak eğim değerleri bulunur. Her kombinasyona ait eğim değerlerinin ortalaması bulunur.
- *DBRANGE*: *DABL* hesaplamasında oluşturulan her kombinasyonun eğim değerlerinden en büyüğünün en küçüğünden çıkarılmasıyla elde edilir.
- *ABDPE*: Dört grup için üç farklı kombinasyon oluşturulur. Bunlar:
  - {1,2}, {3,4}, {1,2,3,4}
  - Her kombinasyona ait gruptaki veriler lineer regresyonda kullanılarak eğim değerleri bulunur. {1,2,3,4} kombinasyonuna ait eğim değerinden {1,2} ve {3,4} kombinasyonlarına ait eğim değerlerinin ortalaması çıkarılır.
- *SASDPE*: Dört grup için üç farklı kombinasyon oluşturulur. Bunlar:
  - {1,2}, {3,4}, {1,2,3,4}
  - Her kombinasyona ait gruptaki veriler lineer regresyonda kullanılarak eğim değerleri bulunur. {1,2,3,4} kombinasyonuna ait eğim değerinden {1,2} ve {3,4} kombinasyonlarına ait eğim değerleri sırayla çıkarılır elde edilen değerlerin mutlağı alınır ve toplanır.

- *SASPE*: Dört grup için iki farklı kombinasyon oluşturulur. Bunlar:
  - {1,2}, {3,4}
  - Her kombinasyona ait gruptaki veriler lineer regresyonda kullanılarak eğim değerleri bulunur. {1,2} ve {3,4} kombinasyonlarına ait eğim değerlerinin mutlak değerleri alınarak toplanır.

Her grafik çeşidine göre hesaplanan parametreler etiketlenerek tek bir veri kümesinde toplanır.

Şekil 3.6'da verilen üçüncü aşamada, veri kümesi CART Karar Ağacı algoritması ile %10 çapraz doğrulama ile eğitilir. Eğitim sonucunda kontrol grafiği örüntü tanınmasında kullanılacak model elde edilir.

## 4. BULGULAR VE TARTIŞMA

Bu bölümde fabrika seviyesinde OTA'ların oluşturduğu trafikte ve OTA seviyesinde konumlandırma sistemlerinde meydana gelebilecek anormal durumların veri odaklı tespiti için önerilen iki farklı yönteme ait bulgular verilmektedir. İlk olarak fabrika seviyesinde OTA'ların oluşturduğu trafikte anormal durumların tespitinde kullanılan denetimli öğrenme yöntemlerine ait sonuçlar verilmektedir. Devamında OTA seviyesinde konumlandırma sistemlerinde zaman içerisinde meydana gelebilecek anormal değişimlerin tespitinde kullanılan pekiştirmeli öğrenme tabanlı anomali tespitinden elde edilen sonuçlar verilmektedir.

### 4.1. Akıllı Fabrikalarda OTA'ların Oluşturduğu Trafikte Anormal Durum Tespiti Sonuçları

Bu bölümde akıllı fabrikalardaki OTA'lar için fabrika seviyesinde OTA'ların oluşturduğu trafikte anormal durumların karar ağacı ile anomali tespiti ve bulanık mantık tabanlı anomali tespiti sonuçları verilmektedir. Karar ağacı yöntemi OTA'ların fabrikada içerisinde kullandığı yollarda meydana gelebilecek; anormal yaya yoğunluğu, doldurma boşaltma noktalarında anormal bekleme, OTA kaynaklı anormal yavaşlama ve OTA kaynaklı nedeni bilinmeyen anormal durumları tespit etmektedir. Bulanık mantık yöntemi ise sadece anomali “var” ve “yok” olarak durum tespiti gerçekleştirmektedir. Anomali nedeni ile ilgili bilgi verilmemektedir. Yöntemlere ait sonuçlar devam eden bölümlerde verilmektedir.

#### 4.1.1. Karar ağacı ile anomali tespiti sonuçları

Birinci yöntemde, sınıflandırma tabanlı karar ağacı algoritması C4.5 veri kümesinde kullanılarak akıllı fabrikalardaki OTA için anomali tespiti gerçekleştirilmiştir. Veri kümesi (Bkz. Bölüm 3.1.2), iki farklı yola ait bilgiler içermektedir. Bunlar; yaya geçidi içeren veya doldurma boşaltma noktası içeren yoldur. Bu yöntemde bu iki farklı yola ait veriler kullanılarak yol parçasının içeriğine göre üç farklı veri kümesi elde edilmiş ve her durum için önerilen eğitim ve test aşamaları gerçekleştirilmiştir. Bu üç farklı durum:

- Sadece yaya geçidi içeren yola ait veriler
- Sadece doldurma boşaltma içeren yola ait bilgiler ve
- Sadece yaya geçidi içeren ve sadece doldurma boşaltma içeren yola ait bilgilerin birlikte olmasıdır.

Etiketli test veri kümesi ve WEKA kullanılarak oluşturulan C4.5 algoritmasına ait model ile C# programlama dilinde geliştirilen uygulamada test edilmiştir. Model kullanılarak test veri kümesindeki veriler için sınıf tahmini elde edilmiştir. Elde edilen sonuçlar C# uygulamasında oluşturulan ara yüzde gösterilmiştir. Şekil 4.1, 4.2 ve 4.3'te sonuçların C# simülasyon ortamı uygulamasında gösterilmesi verilmektedir.



Şekil 4.1. C# Uygulaması simülasyon ortamında normal durum (Örnek vd.'den, 2018)



Şekil 4.2. C# Uygulaması simülasyon ortamında farklı yol parçalarında tespit edilen durma ve yaya yoğunluğu kaynaklı yavaşlama anomalileri (Örnek vd.'den, 2018)



Şekil 4.3. C# Uygulaması simülasyon ortamında farklı yol parçalarında tespit edilen araç kaynaklı bilinmeyen ve yavaşlama anomalileri (Örnek vd.'den, 2018)

Test verilerinin model ile kullanılması sonucunda elde edilen tahminler ve test verisinin gerçek sınıf değeri karşılaştırılarak sistemin doğruluk oranı elde edilir. Çizelge 4.1'de üç farklı durum için elde edilen doğruluk oranları verilmektedir.

Çizelge 4.1. Yol parçasının içeriği ve test doğruluk oranları (Örnek vd. 'den, 2018)

Eğitim ve Test Verisi İçeriği	Algoritma Doğruluk Oranı
Yaya Yolu	%74.8918
Doldurma Boşaltma Noktası	%86.5801
Yaya Yolu ve Doldurma Boşaltma Noktası	%83.7662

#### 4.1.2. Bulanık mantık tabanlı anomali tespiti sonuçları

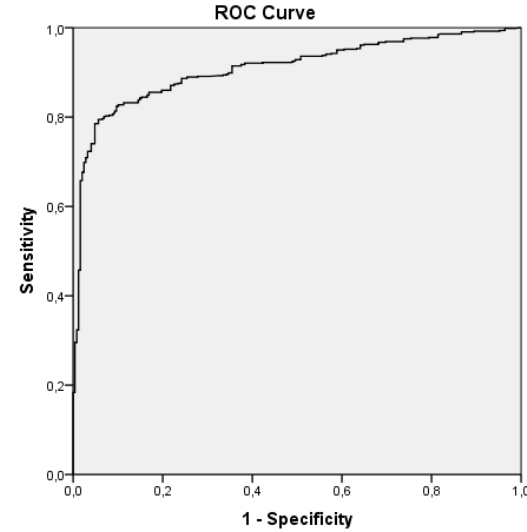
İkinci yöntemde bulanık mantık tabanlı akıllı fabrikalardaki OTA'ların oluşturduğu trafikte oluşabilecek anormal durumların için anomali tespiti gerçekleştirilmiştir. Benzetim ortamında oluşturulan (Bkz. Bölüm 3.1.2) veri kümesindeki tüm öznitelikler kullanılmamıştır. Kullanılan öznitelikler:

- Zaman,
- Ortalama hız,
- İşlem süresi ve
- Yaya yoğunluğudur.

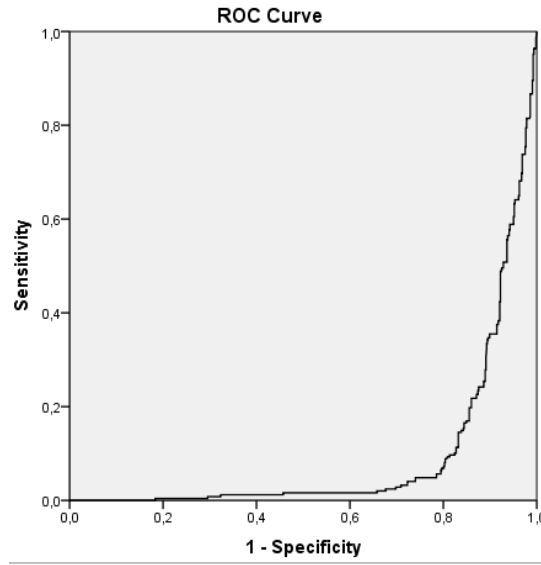
Veri kümesinde bulunan anomali nedeni etiketleri anomali var ise “var” olarak değiştirilmiş anomalinin olmadığı durumları belirten “yok” etiketi ise değiştirilmemiştir. Veri kümesinden yol ve doldurma boşaltma durumlarının aktif olduğu 871 adet veri kullanılmıştır. Bulanık mantık aşamalarının gerçekleştirilmesi ve hesaplamaların yapılması için “Python” kullanılmıştır. Gerekli kodlar Python’da yazılarak veri kümesi için sırasıyla:

- Verilerin üyelik fonksiyonlarına göre bulanıklaştırılması,
- Dilsel değişkenlerin üyelik fonksiyonlarının tanımlanması,
- Kural tablolarının Max-dot yöntemi ile kullanılarak bulanık çıkış değerlerinin elde edilmesi,
- Bulanık çıkış değerlerinin ağırlık merkezi yöntem ile berraklaştırılması ve
- Son olarak anomali olma oranının elde edilmesi yapılmıştır.

Daha sonra verilerin etiket değerleri, anomali durumu (“var” veya “yok”), ve tüm veriler için tek tek hesaplanmış anomali olma oranı “IBM SPSS Statistics 22” programında kullanılarak ROC analizi gerçekleştirilmiştir. Şekil 4.4’te pozitif gerçek durum “var”, Şekil 4.5’te ise pozitif gerçek durum “yok” için ROC analiz sonucunda elde edilen grafikler verilmektedir.



Şekil 4.4. Pozitif gerçek durum “var” için ROC analiz grafiği (Örnek vd.’den, 2020)



Şekil 4.5. Pozitif gerçek durum “yok” için ROC analiz grafiği (Örnek vd.’den, 2020)

ROC analizinden elde edilen grafiklerin sonucunda “4.0” değeri anomali tespiti için uygun eşik değeri olarak seçilmiştir. Eşik değerini bulanık mantık aşamaları sonucunda elde edilen anomali oranı değeri ile kullanarak anomali “var” ve “yok” durumuna karar verilmiştir. Veri kümesindeki verilerin belirlenen eşik değeri ve anomali oranı ile karar verilen anomali durumu “var” ve gerçekteki anomali olma durumu da “var” ise bu durumların yüzdesi doğruluk oranını vermektedir. Ancak veri kümesindeki verilerin belirlenen eşik değeri ve anomali oranı ile karar verilen anomali durumu “yok” ve gerçekteki anomali olma durumu da “var” ise bu durumların oranı ise yanlış-negatif hata oranını vermektedir. Çizelge 4.2’de bulanık mantık tabanlı anomali tespiti sonucunda elde edilen sonuçlar verilmektedir.

Çizelge 4.2. Bulanık mantık tabanlı anomali tespiti değerlendirme metrikleri (Örnek vd.’den, 2020)

Yöntem	Doğruluk Oranı	Yanlış Negatif
Bulanık Mantık	%84,62	0,16



## 4.2. OTA'lar için Konumlandırma Sistemindeki Anormal Durum Tespiti Sonuçları

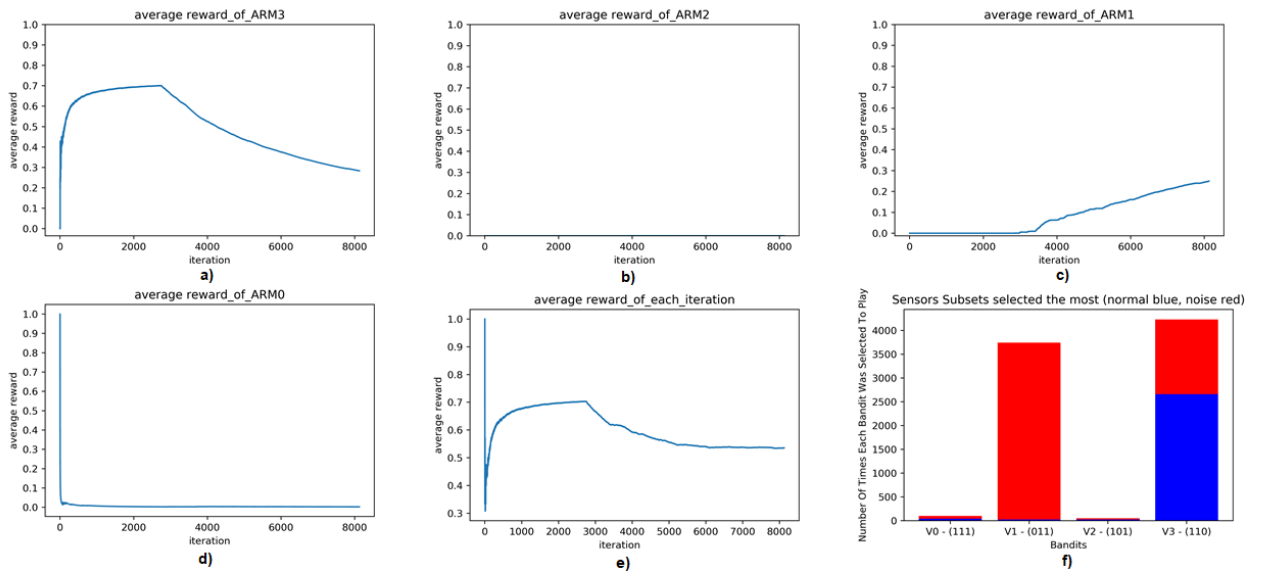
Bu bölümde OTA seviyesinde konumlandırma sistemleri için önerilen veri odaklı pekiştirmeli öğrenme tabanlı anomali tespiti bulguları verilmektedir. Önerilen sistem UCB algoritması, DKG ve CCPR ile OTA için güvenilir konumlandırma sistemleri alt kümesinin karakteristiğinin öğrenilmesi ve karakteristiğın anormal deęişiminin tespitine ait sonuçlar verilmektedir.

Çalışmada, OTA'nın fabrika iç lojistiğindeki malzeme teslim, alma vb. gibi görevlerini yerine getirmeleri sırasında kullandığı önemli bir bileşen olan konumlandırma sistemlerinde zaman içerisinde meydana gelebilecek anormal bir deęişimin tespit edilmesi amaçlanmaktadır. Anormal durumun tespitinde kullanılan veriler IFARLAB test ortamında bulunan yol parçasına aittir. UCB algoritması (Bkz. Algoritma 1) ile bu yol parçası için OTA güvenilir konumlandırma sistemleri alt kümesi, karakteristiğı, öğrenilmiştir. Öğrenme süresi boyunca seçilen her bir kola yani güvenilir konumlandırma sistemleri alt kümesi için aldığı ödül deęerlerinin ortalamaları grafik olarak çizdirilmiştir. Ortalama ödül deęerleri DKG ve CCPR'da kullanılarak sistemin öğrenme sürecinin, ödül deęerlerinin, izlenmesi ve deęişimlerin gözlemlenmesi gerçekleştirilmiştir. Konumlandırma sistemlerinde beklenmedik bir deęişim olması durumunda kollar ve genel ödül grafiklerinde deęişimler oluşmaktadır. Kollara ait ödül grafiklerindeki bu deęişimler izlenerek anormal durumlar tespit edilmiştir. Karakteristikteki bu deęişimleri tespit edebilmek için yapılan testlerde yol için üç farklı veri kümesi kullanılmıştır. Veri kümeleri şu şekildedir:

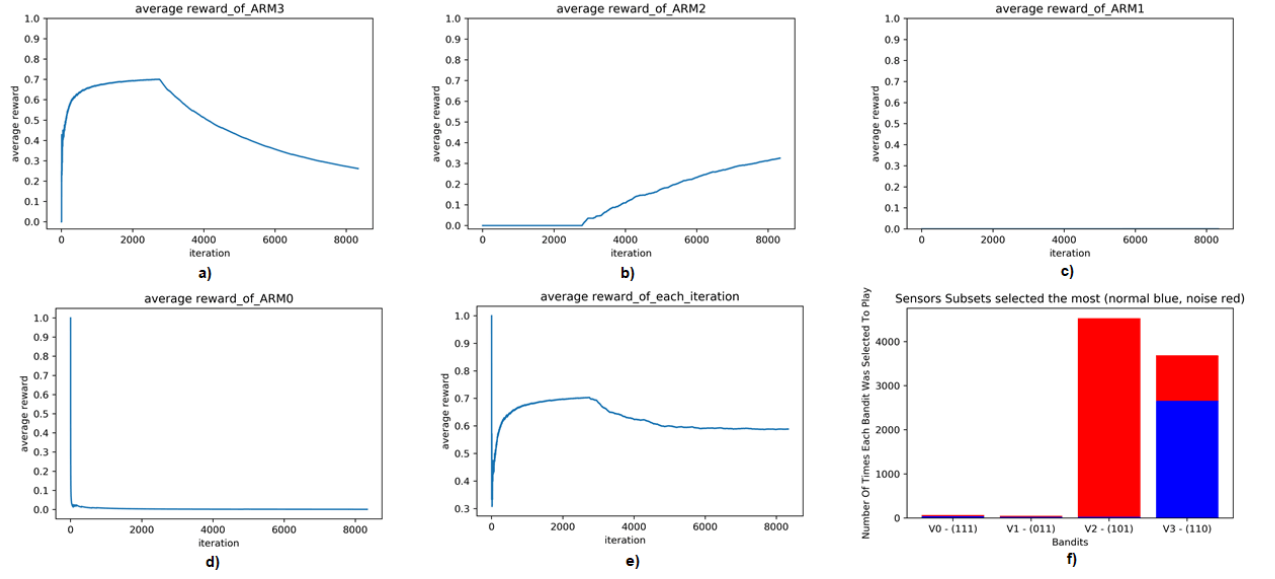
- Test 1: IFARLAB ortamında 500 tur dolaşma için normal konumlandırma sistemleri verisi, sonrasındaki 1000 turda sadece Odometre konumlandırma sistemine gürültü eklenmesi.
- Test 2: IFARLAB ortamında 500 tur dolaşma için normal konumlandırma sistemleri verisi, sonrasındaki 1000 turda sadece AMCL konumlandırma sistemine gürültü eklenmesi.
- Test 3: IFARLAB ortamında 500 tur dolaşma için normal konumlandırma sistemleri verisi, sonrasındaki 1000 turda sadece UWB konumlandırma sistemine gürültü eklenmesi.

Seçilen yol için üç farklı veri seti kullanılarak yapılan deney sonucunda elde edilen her kol için genel ortalama ödül değerlerine ait grafikler Şekil 4.6, 4.7 ve 4.8’de verilmektedir. Şekillerde yer alan “*KOL3*, *KOL2*, *KOL1*, *KOL0*” ve “*V3*, *V2*, *V1*, *V0*” konumlandırma sistemleri alt kümelerinin (kollar) açıklamaları şu şekildedir:

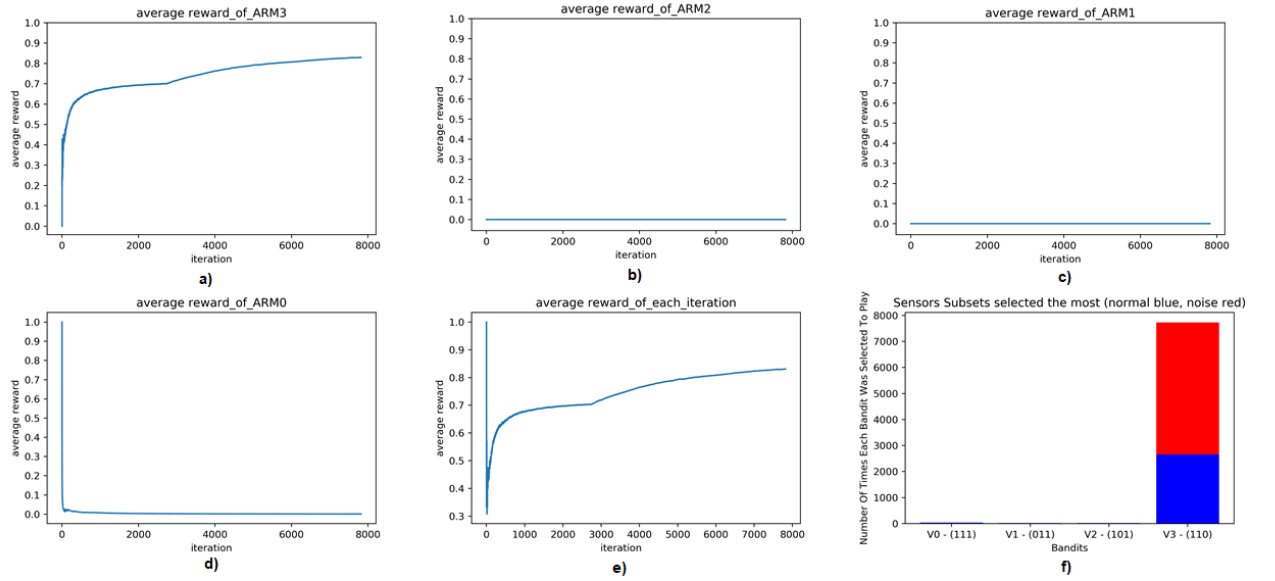
- *KOL3* ve *V3*: UWB konumlandırma sistemine güvenilmediği durum
- *KOL2* ve *V2*: AMCL konumlandırma sistemine güvenilmediği durum
- *KOL1* ve *V1*: Odometre konumlandırma sistemine güvenilmediği durum
- *KOL0* ve *V0*: Tüm konumlandırma sistemlerine güvenildiği durum



Şekil 4.6. Test 1 için UCB algoritmasında elde edilen grafikler, (a) *KOL3* için ortalama ödül grafiği, (b) *KOL2* için ortalama ödül grafiği, (c) *KOL1* için ortalama ödül grafiği, (d) *KOL0* için ortalama ödül grafiği, (e) genel ortalama ödül grafiği, (f) *KOL0*, *KOL1*, *KOL2* ve *KOL3*’ün seçilme sayılarının normal ve gürültülü veriye göre değişimi (normal veri mavi renk, gürültülü veri kırmızı renk)



Şekil 4.7. Test 2 için UCB algoritmasında elde edilen grafikler, (a) *KOL3* için ortalama ödül grafiği, (b) *KOL2* için ortalama ödül grafiği, (c) *KOL1* için ortalama ödül grafiği, (d) *KOL0* için ortalama ödül grafiği, (e) genel ortalama ödül grafiği, (f) *KOL0*, *KOL1*, *KOL2* ve *KOL3*'ün seçilme sayılarının normal ve gürültülü veriye göre değişimi (normal veri mavi renk, gürültülü veri kırmızı renk)

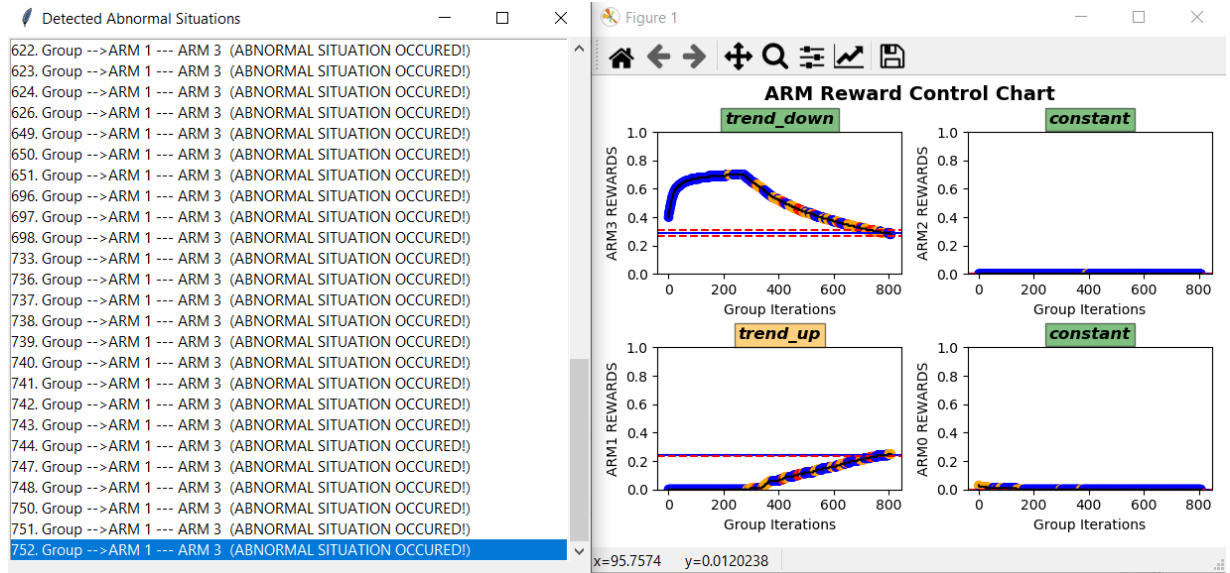


Şekil 4.8. Test 3 için UCB algoritmasında elde edilen grafikler, (a) *KOL3* için ortalama ödül grafiği, (b) *KOL2* için ortalama ödül grafiği, (c) *KOL1* için ortalama ödül grafiği, (d) *KOL0* için ortalama ödül grafiği, (e) genel ortalama ödül grafiği, (f) *KOL0*, *KOL1*, *KOL2* ve *KOL3*'ün seçilme sayılarının normal ve gürültülü veriye göre değişimi (normal veri mavi renk, gürültülü veri kırmızı renk)

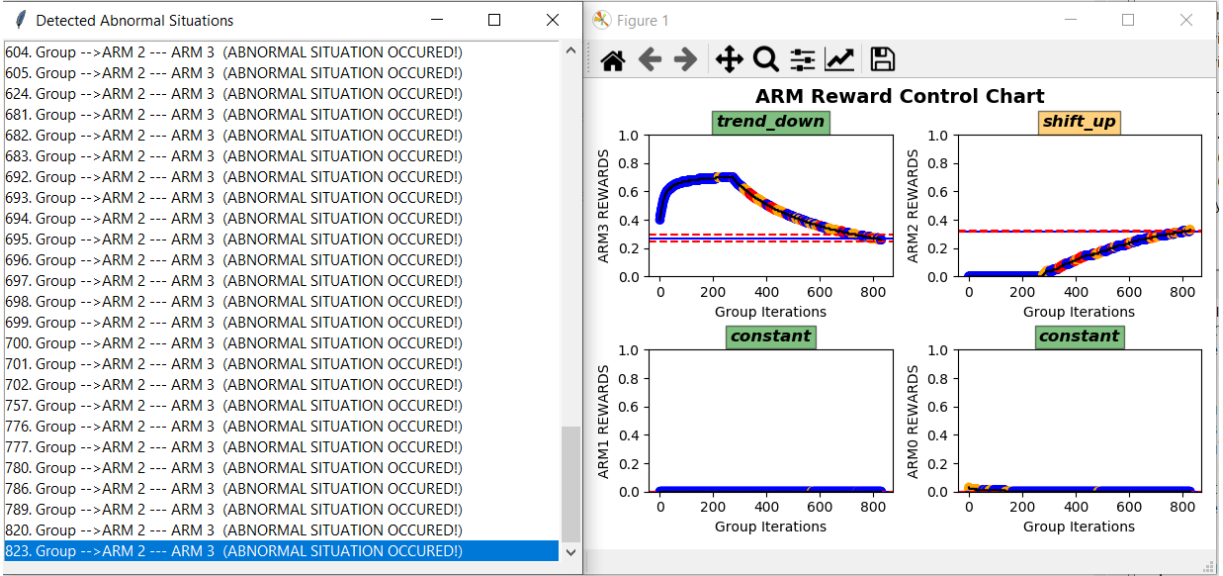
Şekillerden de görüldüğü üzere normal veri süresince öğrenilen karakteristikten sonra gürültülü veri gelmeye başladığında kollara ait ortalama ödül grafiklerinde

değişiklikler oluşmaya başlamıştır. Örnek olarak ortalama ödül değerlerinin karakteristiğinin öğrenilen kolda düşmeye başlaması ile yeni güvenilir konumlandırma sistemleri alt kümesi olarak seçilen kolda ise artmaya başlaması verilebilir. Bu ödül değerlerindeki değişimlerin analiz edilerek anomali tespitinin gerçekleştirilebilmesi için “DKG” ve “CCPR” kullanılarak anormal karakteristikler tespit edilmiş ve tespit edildiği grup ve hangi kollar arasında olduğu yazdırılmış, ödül grafiklerinden elde edilen DKG’leri her bir kol için her adımda çizdirilmiş ve her adımda tespit edilen örüntü bilgisi çizdirilen grafiklerde yazdırılmıştır. Şekil 4.9, 4.10 ve 4.11’de seçilen yol için üç farklı senaryonun testi sonucunda elde edilen grafikler ve tespit edilen anormal durumların hangi kollar arasında olduğu bilgileri verilmektedir. Grafiğin üstündeki alanda verilen örüntü bilgisinin renkleri üç çeşittir ve açıklamaları şu şekildedir:

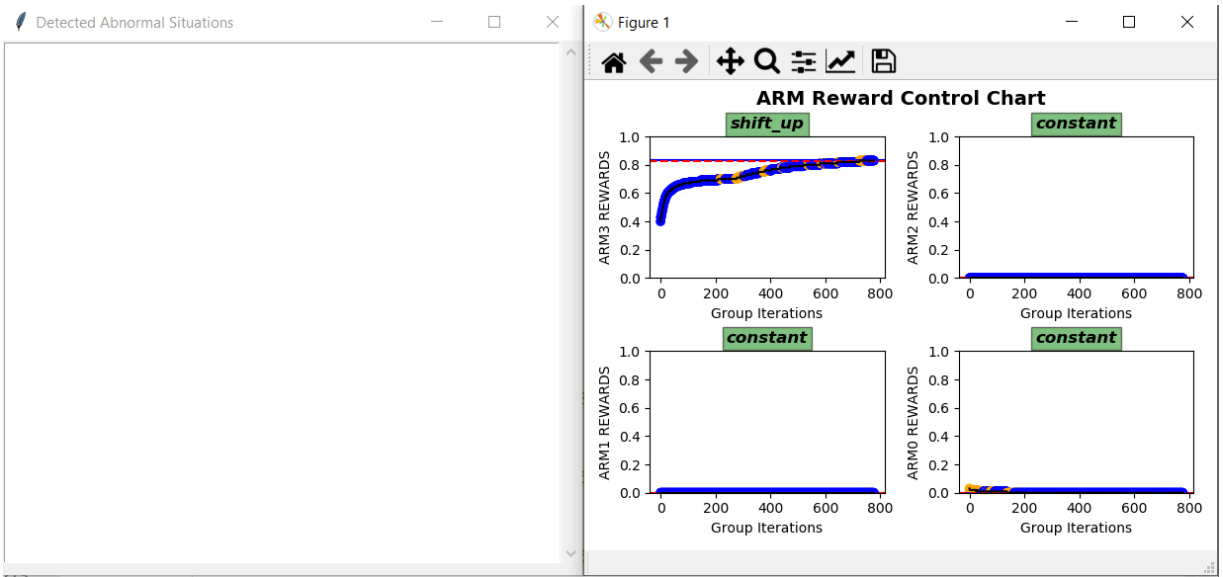
- Yeşil: o kol için durumun normal olduğu belirtir.
- Sarı: o kol için durumun uyarı olduğu belirtir.
- Kırmızı: o kol ve sistem için anormal bir durum olduğu belirtir. Anomali tespiti gerçekleşir.



Şekil 4.9. Test 1 için önerilen sistem ile yapılan anomali tespiti



Şekil 4.10. Test 2 için önerilen sistem ile yapılan anomali tespiti



Şekil 4.11. Test 3 için önerilen sistem ile yapılan anomali tespiti

Pekiştirmeli öğrenme tabanlı önerilen anomali tespit sistemi akıllı fabrikalardaki OTA için OTA seviyesinde konumlandırma sistemlerinde meydana gelebilecek anormal durumların çevrim içi olarak tespiti için tasarlanmıştır. Tasarlanan sistem OTA konumlandırma sistemlerini OTA'ların aktif olarak faaliyet gösterdiği sürece izleyerek sistem karakteristiğini öğrenir. Öğrenilen karakteristikte meydana gelebilecek olası değişimleri, anomalileri dinamik olarak tespit etmektedir. Üç farklı senaryo kullanılarak elde edilen test sonuçlarında önerilen sistemin konumlandırma sistemlerinden elde edilen

verilerde meydana gelebilecek beklenmedik anormallikleri konumlandırma sistemleri arasındaki ilişkiyi de göz önünde bulundurarak başarıyla tespit etmiştir.

## 5. SONUÇ VE ÖNERİLER

Endüstri 4.0 ve akıllı fabrika alanında gelişmeler ile fabrika içinde taşıma görevleri için OTA kullanımını yaygınlaştırmıştır. Ancak OTA çalışmasını olumsuz etkileyebilecek durumlarda insan müdahalesi olmaksızın veri üzerinden bu sorunların tespit edilmesi gereklidir. Ancak bu anormal durumlar çeşitli sebeplerden dolayı meydana gelebilir ve tam tanımlamaların yapılması mümkün olmayabilir. Bu nedenle akıllı fabrikalardaki OTA'lar için fabrika ve OTA seviyesinde veri odaklı denetimli öğrenme ve pekiştirmeli öğrenme kullanılarak anomali tespiti gerçekleştirilmiştir.

Denetimli öğrenme ile anomali tespitinin gerçekleştirilmesi için verinin etiketlenmesi de gereklidir. Ancak bu işlem ek süre harcanmasına ve öğrenilen durumlar dışındaki anormalliklerin kaçırılmasına neden olmaktadır. Diğer taraftan pekiştirmeli öğrenme bu açıdan daha avantajlıdır. Pekiştirmeli öğrenme ile anomali tespitinde veri üretimi gerçekleştirildikten sonra verinin etiketlenmesi gerekmemektedir. Bunun nedeni pekiştirmeli öğrenme mantığından kaynaklanmaktadır. Pekiştirmeli öğrenme veri kümesinden karakteristiği öğrenir. Öğrenilen karakteristiğe aykırı durumların oluşması ise anormal durum olarak değerlendirilebilir.

Veri odaklı anomali tespiti, sistemlerin izlenebilmesi ve anormal durumların erken tespiti için önemlidir. Gelecek çalışmalarda OTA'lar için anomali tespiti kapsamında farklı öğrenme yöntemlerinden derin pekiştirmeli öğrenme kullanılabilir. Sadece OTA değil, fabrikalardaki farklı sistemlerin bir bütün olarak değerlendirilerek anormal durumların tespit edilmesi üzerine çalışmalar gerçekleştirilebilir.

## KAYNAKLAR DİZİNİ

- Agrawal, R., 1995, Sample mean-based index policies with  $O(\log n)$  regret for the multi-armed bandit problem, *Advances in Applied Probability*, 27, 1054–1078.
- Agrawal, S., Goyal, N., 2012, Analysis of thompson sampling for the multi-armed bandit problem, In *Conference on learning theory*, pp. 39-1.
- Akiyama, Y., Kasai, Y., Iwata, M., Takahashi, E., Sato, F., Murakawa, M., 2015, Anomaly detection of solar power generation systems based on the normalization of the amount of generated electricity, In *2015 IEEE 29th International Conference on Advanced Information Networking and Applications*, 294-301.
- Auer, P., Cesa-Bianchi, N., Fischer, P., 2002, Finite-time analysis of the multiarmed bandit problem, *Machine learning*, 47(2-3), 235-256.
- Banerjee, A., Chandola, V., Kumar, V., Srivastava, J., Lazarevic, A., 2008, Anomaly detection: A tutorial, <https://www.siam.org/meetings/sdm08/TS2.ppt>, erişim tarihi: 18.11.2020.
- Banjanovic-Mehmedovic, L., Hajdarevic, A., Kantardzic, M., Mehmedovic, F., Dzananovic, I., 2017, Neural network-based data-driven modelling of anomaly detection in thermal power plant, *Automatika: časopis za automatiku, mjerenje, elektroniku, računarstvo i komunikacije*, 58(1), 69-79.
- Barraza-Urbina, A., Glowacka, D., 2020, Introduction to Bandits in Recommender Systems, In *Fourteenth ACM Conference on Recommender Systems*, 748-750.
- Bouneffouf, D., Rish, I., 2019, A survey on practical applications of multi-armed and contextual bandits, *arXiv preprint arXiv:1904.10040*.
- Chandola, V., Banerjee, A., Kumar, V., 2007, Outlier detection: A survey. *ACM Computing Surveys*, 14, 15.
- Chandola, V., Banerjee, A., Kumar, V., 2009, Anomaly detection, *ACM Computing Surveys*, 41(3), 1–58, doi: 10.1145/1541880.1541882.
- Değirmenci E., Örnek Ö., Yazıcı A., 2020, İçeriksel Çok Kollu Slot Makinesi ile Akıllı Fabrikalarda Trafik Karakteristiklerinin Öğrenilmesi ve Anomali Tespiti, 28. IEEE Sinyal İşleme ve İletişim Uygulamaları Kurultayı, IEEE (yayınlanmamış).
- Ding, K., Li, J., Liu, H., 2019, Interactive anomaly detection on attributed networks, In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, 357-365.



**KAYNAKLAR DİZİNİ (devam)**

- Duc, T. L., Leiva, R. G., Casari, P., Östberg, P. O., 2019, Machine learning methods for reliable resource provisioning in edge-cloud computing: A survey, *ACM Computing Surveys (CSUR)*, 52(5), 1-39.
- Ebrahimzadeh, A., Ranaee, V., 2009, A hybrid intelligent technique for recognition of control chart patterns, In 2009 First International Conference on Networked Digital Technologies, 32-36, IEEE.
- ESOGÜ, 2020, Eskisehir Osmangazi University Intelligent Factory and Robotics Laboratory, <https://ifarlab.ogu.edu.tr>, erişim tarihi: 13.12.2020.
- Evangelou, M., Adams, N. M., 2020, An anomaly detection framework for cyber-security data, *Computers & Security*, 97, 101941.
- Fahim, M., & Sillitti, A., 2019, Anomaly detection, analysis and prediction techniques in iot environment: A systematic literature review, *IEEE Access*, 7, 81664-81681.
- Ferdowsi, A., Ali, S., Saad, W., Mandayam, N. B., 2019, Cyber-physical security and safety of autonomous connected vehicles: Optimal control meets multi-armed bandit learning, *IEEE Transactions on Communications*, 67(10), 7228-7244.
- Freeman, P., Pandita, R., Srivastava, N., Balas, G. J., 2013, Model-based and data-driven fault detection performance for a small UAV, *IEEE/ASME Transactions on mechatronics*, 18(4), 1300-1309.
- Fuqua, D., Razzaghi, T., 2020, A cost-sensitive convolution neural network learning for control chart pattern recognition, *Expert Systems with Applications*, 150, 113275.
- Gauri, S. K., Chakraborty, S., 2006, Feature-based recognition of control chart patterns, *Computers & Industrial Engineering*, 51(4), 726-742.
- Gauri, S. K., Chakraborty, S., 2007, A study on the various features for effective control chart pattern recognition, *The International Journal of Advanced Manufacturing Technology*, 34(3-4), 385-398.
- Gauri, S. K., 2010, Control chart pattern recognition using feature-based learning vector quantization, *The International Journal of Advanced Manufacturing Technology*, 48(9-12), 1061-1073.
- Ghobakhloo, M., 2020, Industry 4.0, digitization, and opportunities for sustainability, *Journal of Cleaner Production*, 252, 119869.

**KAYNAKLAR DİZİNİ (devam)**

- Guh, R. S., 2005, A hybrid learning-based model for on-line detection and analysis of control chart patterns, *Computers & Industrial Engineering*, 49(1), 35-62.
- Guh, R. S., Shiue, Y. R., 2005, On-line identification of control chart patterns using self-organizing approaches, *International Journal of Production Research*, 43(6), 1225-1254.
- Guh, R. S., 2008, Real-time recognition of control chart patterns in autocorrelated processes using a learning vector quantization network-based approach, *International Journal of Production Research*, 46(14), 3959–3991.
- Guh, R. S., 2010, Simultaneous process mean and variance monitoring using artificial neural networks, *Computers and Industrial Engineering*, 58(4), 739–753.
- Hachicha, W., Ghorbel, A., 2012, A survey of control-chart pattern-recognition literature (1991–2010) based on a new conceptual classification scheme, *Computers & Industrial Engineering*, 63(1), 204-222.
- Hayes, M. A., Capretz, M. A. M., 2014, Contextual Anomaly Detection in Big Sensor Data, 2014 IEEE International Congress on Big Data, Presented at the 2014 IEEE International Congress on Big Data (BigData Congress), 64-71, <https://doi.org/10.1109/bigdata.congress.2014.19>.
- Hoi, S. C., Sahoo, D., Lu, J., Zhao, P., 2018, Online learning: A comprehensive survey, arXiv preprint arXiv:1802.02871.
- Huang, S., Guo, Y., Yang, N., Zha, S., Liu, D., Fang, W., 2020, A weighted fuzzy C-means clustering method with density peak for anomaly detection in IoT-enabled manufacturing process, *Journal of Intelligent Manufacturing*, 1-17.
- Kaelbling, L. P., Littman, M. L., Moore, A. W., 1996, Reinforcement Learning: A Survey, *Journal of Artificial Intelligence Research*, 4, 237–285, doi: 10.1613/jair.301
- Khalastchi, E., Kalech, M., Kaminka, G. A., Lin, R., 2015, Online data-driven anomaly detection in autonomous robots, *Knowledge and Information Systems*, 43(3), 657-688.
- Kovářík, M., Klímek, P., 2012, The usage of time series control charts for financial process analysis, *Journal of Competitiveness*.
- Liang, W., Pu, X., Xiang, D., 2017, A distribution-free multivariate CUSUM control chart using dynamic control limits, *Journal of Applied Statistics*, 44(11), 2075-2093.

**KAYNAKLAR DİZİNİ (devam)**

- Liang, Y. C., Wang, S., Li, W. D., Lu, X., 2019, Data-driven anomaly diagnosis for machining processes, *Engineering*, 5(4), 646-652.
- Losada, D. E., Parapar, J., Barreiro, A., 2017, Multi-armed bandits for adjudicating documents in pooling-based evaluation of information retrieval systems, *Information Processing & Management*, 53(5), 1005-1025.
- Lu, Z., Wang, M., Dai, W., 2020, A condition monitoring approach for machining process based on control chart pattern recognition with dynamically-sized observation windows, *Computers & Industrial Engineering*, 142, 106360.
- Macdonald, A. J., Lim, M., Prystay, E., Matasci, G., Martin-Boyd, L., Webster, A., Busler, J., 2020, Unsupervised behaviour anomaly detection from fixed camera full motion video, In *Artificial Intelligence and Machine Learning in Defense Applications II*, International Society for Optics and Photonics.
- Maimon, O., Rokach, L. (Eds.), 2005, *Data mining and knowledge discovery handbook*, Springer, Boston, MA, <https://doi.org/10.1007/978-0-387-09823-4>, p.133-265.
- Mehrotra, K. G., Mohan, C. K., Huang, H., 2017, *Anomaly detection principles and algorithms*, Springer International Publishing, New York, NY, USA, p.1-217.
- Mendel, J. M., 1995, Fuzzy logic systems for engineering: a tutorial, *Proceedings of the IEEE*, 83(3), 345-377. doi: <https://doi.org/10.1109/5.364485>.
- Montgomery, D. C., 2001, *Introduction to statistical quality control*, New York: Wiley.
- Montgomery, D. C., 2009, *Introduction to statistical quality control*, sixth edition, John Wiley & Sons, p.177-433.
- Nguyen, L. V., Kapinski, J., Jin, X., Deshmukh, J. V., Butts, K., Johnson, T. T., 2017, Abnormal data classification using time-frequency temporal logic, In *Proceedings of the 20th international conference on hybrid systems: Computation and control*, 237-242.
- Örnek Ö., Vatan S., Sarıoğlu S., Yazıcı A., 2018, Anomaly Detection for Autonomous Transfer Vehicles in Smart Factories, 2018 6th International Conference on Control Engineering & Information Technology (CEIT), Istanbul, Turkey, 2018, 1-5, doi: [10.1109/CEIT.2018.8751935](https://doi.org/10.1109/CEIT.2018.8751935).

**KAYNAKLAR DİZİNİ (devam)**

- Örnek Ö., Gülbandılar E., Yazıcı A., 2020, Akıllı Fabrikalardaki Otonom Taşıyıcılar için Bulanık Mantık Tabanlı Anomali Tespiti, Eskişehir Osmangazi Üniversitesi Mühendislik ve Mimarlık Fakültesi Dergisi, 28 (1), 53-61, DOI: 10.31796/ogummf.619239.
- Osterrieder, P., Budde, L., 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.
- Pan, D., 2017, Hybrid data-driven anomaly detection method to improve UAV operating reliability, In 2017 Prognostics and System Health Management Conference (PHM-Harbin), 1-4, IEEE.
- Parkhideh, B., Case, K. E., 1989, The economic design of a dynamic X-control chart, IIE transactions, 21(4), 313-323.
- Pham, D. T., Wani, M. A., 1997, Feature-based control chart pattern recognition, International Journal of Production Research, 35(7), 1875–1890.
- Prabhu, G. R., Bhashyam, S., Gopalan, A., Sundaresan, R., 2019, Learning to Detect an Anomalous Target with Observations from an Exponential Family, In DSW, 88-92.
- Roberts, S., 2020, The Upper Confidence Bound (UCB) Bandit Algorithm, <https://towardsdatascience.com/the-upper-confidence-bound-ucb-bandit-algorithm-c05c2bf4c13f>, erişim tarihi: 20.11.2020.
- Sarmadi, H., Karamodin, A., 2020, A novel anomaly detection method based on adaptive Mahalanobis-squared distance and one-class kNN rule for structural health monitoring under environmental effects, Mechanical Systems and Signal Processing, 140, 106495.
- Savchenko, A. V., Milov, V. R., 2017, Decision Support in Intelligent Maintenance-planning Systems Based on Contextual Multi-armed Bandit Algorithm, Procedia Computer Science, 103(C), 316-323.
- Shen, W., Wang, J., Jiang, Y. G., Zha, H., 2015, Portfolio choices with orthogonal bandit learning, In Twenty-fourth international joint conference on artificial intelligence.
- Slivkins, A., 2019, Introduction to multi-armed bandits, arXiv preprint arXiv:1904.07272.
- Song, J., Vorburger, T., 2007, Verifying measurement uncertainty using a control chart with dynamic control limits, NCSLI Measure, 2(3), 76-80.

**KAYNAKLAR DİZİNİ (devam)**

- Stojanovic, L., Dinic, M., Stojanovic, N., Stojadinovic, A., 2016, Big-data-driven anomaly detection in industry (4.0): An approach and a case study, In 2016 IEEE International Conference on Big Data (Big Data), 1647-1652, IEEE.
- Stojanovic, N., Dinic, M., Stojanovic, L., 2017, A data-driven approach for multivariate contextualized anomaly detection: Industry use case, In 2017 IEEE International Conference on Big Data (Big Data), 1560-1569, IEEE.
- Sutton, S. R., Barto, G. A., 2018, Reinforcement Learning: An Introduction, MIT Yayıncılık, p.1-179.
- Swift, J. A., 1987, Development of a knowledge based expert system for control chart pattern recognition and analysis (Doctoral dissertation, Oklahoma State University).
- Vanhoeyveld, J., Martens, D., Peeters, B., 2020, Value-added tax fraud detection with scalable anomaly detection techniques, Applied Soft Computing, 86, 105895.
- Wang, C. H., Guo, R. S., Chiang, M. H., Wong, J. Y., 2008, Decision tree based control chart pattern recognition, International Journal of Production Research, 46(17), 4889–4901.
- Wang, C. H., Dong, T. P., Kuo, W., 2009, A hybrid approach for identification of concurrent control chart patterns, Journal of Intelligent Manufacturing, 20(4), 409–419.
- Wang, P., Govindarasu, M., Ashok, A., Sridhar, S., McKinnon, D., 2017, Data-driven anomaly detection for power system generation control, In 2017 IEEE International Conference on Data Mining Workshops (ICDMW), 1082-1089, IEEE.
- Wani, M. A., Rashid, S., 2005, Parallel algorithm for control chart pattern recognition, In Fourth International Conference on Machine Learning and Applications (ICMLA'05), IEEE.
- Wasserman, G. S., 1994, Short run SPC using dynamic control chart, Computers & industrial engineering, 27(1-4), 353-356.
- Wu, C. W., Chen, M., 2018, Early anomaly detection in wind turbine bolts breaking problem—Methodology and application, In 2018 IEEE 3rd International Conference on Big Data Analysis (ICBDA), 402-406.
- Yan, W., 2016, One-class extreme learning machines for gas turbine combustor anomaly detection, In 2016 international joint conference on neural networks (ijcnn), IEEE, 2909-2914.

**KAYNAKLAR DİZİNİ (devam)**

- Zadeh, L. A., 1988, Fuzzy logic, In Computer, 21, 4, 83-93. doi: <https://doi.org/10.1109/2.53>.
- Zang, D., Liu, J., Wang, H., 2018, Markov chain-based feature extraction for anomaly detection in time series and its industrial application, In 2018 Chinese Control And Decision Conference (CCDC), 1059-1063.
- Zhao, P., Kurihara, M., Tanaka, J., Noda, T., Chikuma, S., Suzuki, T., 2017, Advanced correlation-based anomaly detection method for predictive maintenance, In 2017 IEEE International Conference on Prognostics and Health Management (ICPHM), 78-83.
- Zhou, T., Wang, Y., Tan, Y., 2020, Spoiled for Choice? Personalized Recommendation for Healthcare Decisions: A Multi-Armed Bandit Approach, arXiv preprint arXiv:2009.06108.
- Zhuang, H., Wang, C., Wang, Y., 2017, Identifying outlier arms in multi-armed bandit, In Advances in Neural Information Processing Systems, 5204-5213.
- Zorriassatine, F., Al-Habaibeh, A., Parkin, R. M., Jackson, M. R., Coy, J., 2005, Novelty detection for practical pattern recognition in condition monitoring of multivariate processes: A case study, International Journal of Advanced Manufacturing Technology, 25(9–10), 954–963.