Kümeleme Algoritmaları Kullanılarak Mobil Robot Verileri Üzerinde Anomali Tespiti

Zekeriyya Demirci, Yunus Sabri Kırca, Dr. Eyüp Çınar
Eskişehir Osmangazi Üniversitesi Bilgisayar Mühendisliği

Özet—Endüstri 4.0 ile beraber gittikçe gelişen ve daha akıllı hale gelen sistemler anlık olarak sürekli veriler üretmektedir. Bu veriler, sistem bileşenlerinde bulunan çeşitli sensöreler tarafından üretilmektedir. Zamanla çok büyük boyutlara ulaşan bu verilerin sistemi geri beslemesi için analiz edilmesi ve anlamlı sonuçlar üretilmesi gerekmektedir. Verilere bakarak olası hataların tahmini veya tespit edilmesi, üretim ve bakım maliyetlerinin düşürülmesi vb. birçok konuda sistemde geliştirmeler yapmam mümkündür. Bu çalışmada, Otonom Taşıyıcı Araç(OTA)'ta bulunan akım ve titreşim sensörlerinden toplanan veriler üzerinde k-means ve hiyerarşik kümeleme metotları kullanılarak anomali tespiti yapılmıştır. Her iki yöntem de uygulanarak performans metrikleri oluşturulmuştur. Kümeleme sonuçlarına bakılarak veriler üzerinde anomali skorları bulunmuş ve olası anomaliler için bir oran belirlenmiştir. Bu çalışmanın amacı, mobil robotların hareketleri esnasında toplanan sensör verilerinden araçta olası bir anomalinin tespiti ve bu anomalinin beklenen bir davranış olup olmadığının gözlemlenmesidir.

***Anahtar Kelimeler*— Makine öğrenmesi, kümeleme, k-means, hiyerarşik kümeleme, anomali tespiti, mobil robotlar, endüstriyel robotlar.**

I. GİRİŞ

ENDÜSTRİ 4.0 ile beraber fiziksel ve dijital sistemler arasındaki entegrasyon kaçınılmaz hale gelmiştir. Bu ortamların entegrasyonu ile beraber farklı ekipmanlardan büyük miktarda veri toplanabilir hale gelmiştir[1]. Endüstriyel sistemler tarafından toplanan büyük miktarda veri, bir endüstriyel üretim hattında meydana gelen süreçler, olaylar ve alarmlar hakkında bilgi içerir. Bu veriler işlendiğinde ve analiz edildiğinde üretim sürecinden ve sistem dinamiklerinden değerli bilgiler ortaya çıkarılabilir. Verilere dayalı analitik yaklaşımlar uygulayarak bakım maliyetlerini düşürme, makine arızalarını azaltma ve yedek parça ömrünü artırma gibi konularda önceden tahminler yapmak mümkündür[2,3,4]. Zamanla çok büyük boyutlara ulaşan bu verilerden bir değer üretilmesi ve verinin sisteme geri kazandırılması gerekmektedir.

Endüstriyel robotların daha akıllı hale gelmesiyle birlikte ürettikleri veri miktarı da artmıştır. Bu veriler çeşitli sensörlerden sürekli olarak toplanmaktadır. Dolayısıyla böylesine büyük miktardaki verinin anlam kazanması ve sistemi geri beslemesi için işlenmesi gerekmektedir. Burada ise devreye makine öğrenmesi algoritmaları girmektedir. Makine öğrenmesi temelde gözetimli öğrenme ve gözetimsiz öğrenme olarak ikiye ayrılır. Gözetimli öğrenmede geçmiş zamana ait verilere ihtiyaç duyulur. Bu veriler etiketlenerek modeller eğitilir ve yeni gelen veri bu model üzerinden işlenir. Ses tanıma, nesne tanıma vb. birçok alanda kullanılmaktadır. [5] Gözetimli öğrenmede karar destek makinaları, lojistik regresyon, karar ağaçları, k-en yakın komşu algoritması gibi çeşitli algoritmalar bulunur. Gözetimsiz öğrenmede ise veriler etiketlenmeden kullanılır ve veri üzerinden şablonlar çıkarmaya çalışır. Gözetimsiz öğrenme algoritmalarında en yaygın kullanılan yöntem de kümelemedir[5].

Kümeleme, verilen veri setini gruplara ayıran bir makine öğrenmesi tekniğidir. Aynı gruptaki verilerin benzer özellik veya davranışa sahip verileri içermesi beklenirken farklı gruplardaki veriler birbirinden daha farklı özellikler bulundurmalıdır. K-means ve Hiyerarşik kümeleme en yaygın kullanılan kümeleme algoritmalarındandır. Kümeleme algoritmaları pazarlama, endüstri, bilgisayarla görü vb. birçok alanda kullanılmaktadır[6].

Endüstriyel robotlardan ve cihazlardan toplanan veriler kümeleme algoritmaları kullanılarak analiz edilebilir ve sistem hakkında bize bilgi verebilir. Bu noktada, sistemde meydana gelebilecek hataların tespiti, öngörülmesi ve önlenmesi konularında çalışmalar yapılmaktadır[7]. Veride bulunan beklenmeyen davranışlar anomali olarak adlandırılır. Bu anomaliler kümeleme algoritmaları kullanılarak bulunabilir. Anomali tespiti, beklenen davranışa uymayan veri kalıplarını bulma sürecini ifade eder[8]. Bu uyumsuz kalıplar genellikle farklı uygulama alanlarında anormallikler, aykırı değerler, uyumsuzluklar veya istisnalar olarak adlandırılır[8]. Anomali tespiti sağlık hizmetleri, akıllı cihazlar, akıllı şehirler, nesnelerin interneti, dolandırıcılık algılama, bulut gibi çok çeşitli uygulamalarda yaygın kullanım bulmaktadır[8].

Bu çalışmada, Otonom Taşıyıcı Araç (OTA)'dan alınan akım ve titreşim verileri üzerinden hiyerarşik kümeleme ve k-means kümeleme algoritmaları kullanılarak anomali tespiti yapılmaktadır. Özellik çıkarma(Feature Extraction) ve dinamik zaman atlama(Dynamic Time Warping) yöntemleri kullanılarak her iki kümeleme algoritması üzerinde de sonuçlar elde edilmiştir.

II. LİTERATÜR ÖZETİ

Endüstride üretilen verinin artmasıyla beraber verilerin sisteme geri kazandırılması için çalışmalar başlamıştır. Yeterli büyüklükte veriler işlenerek sistem bakım maliyetlerini düşürme, makine arızalarını azaltma ve yedek parça ömrünü

artırma gibi konularda önceden tahminler yapmak mümkün olmaya başlamıştır[2,3,4]. Makine öğrenmesi algoritmaları bu veriler üzerinde koşturularak sonuçlar elde edilmeye başlanmıştır. Makine öğrenmesi literatürde temel olarak denetimli ve denetimsiz olarak ikiye ayrılmıştır[5,7,9]. Denetimsiz öğrenmede en yaygın kullanılan teknik kümeleme algoritmalarıdır[5]. Kümeleme, verileri benzerliklerine göre kümelere ayırma sürecini ifade eder[5]. Benzerlikler, kümeler arasındaki mesafe ile ilişkilidir[5]. Kümele algoritmalarından k-means ve hiyerarşik kümeleme en yaygın kullanılan yöntemlerdir[6]. Kümeleme yöntemleriyle veriler üzerinde analizler yapmak mümkündür. Bu sayede verilerde olası hataların tespiti ve tahmin edilmesi gibi çalışmalar mevcuttur. Bu hatalar literatürde anomali olarak adlandırılmaktadır. Anomali tespiti, veri kümelerindeki beklenmedik öğeleri veya olayları belirleme sürecidir ve genelde etiketlenmemiş verilere uygulanır. Birçok pratik uygulamada; ağa izinsiz giriş, dolandırıcılık tespiti ve birçok bilim dalında kullanılır[10]. Özellikle kablosuz iletişim, yerel ağ uygulamalarında yapılmış çalışmalar da mevcuttur[11,12]. Endüstride de bakım maliyetlerini düşürme, makine arızalarını azaltma ve yedek parça ömrünü artırma gibi konularda çalışmalar mevcuttur[2,3,4]. Bunun yanı sıra anomali tespiti üzerine de çalışmalar mevcuttur. Hiyerarşik kümeleme; kredi kartı verileri[13], zaman serisi verileri[14,15] ve araç güzergah verileri[16] üzerine yapılan anomali tespiti çalışmalarında da kullanılmıştır. Endüstriyel pompalar[17], elektrik motorları [18], endüstriyel kontrol sistemleri üzerine çalışmalar da mevcuttur. Ayrıca endüstriyel motorlardan toplanan sensör verileri kullanılarak yapılan anomali tespiti çalışmaları da mevcuttur[19].

K-means kümeleme algoritması, veri setinde bulunan her bir verinin sayısal değerlerini koordinat olarak kabul eden ve bu koordinatları baz alarak verileri uzaklıklarına göre kümeleyen bir algoritmadır[20]. Literatürde AS 136 olarak da adı geçmektedir. K-means kümeleme gerçekleştirilen her iterasyonda daha iyi sonuçlar elde etmek üzere çalışan bir algoritmadır fakat belirlenen K noktalarının konumları başlangıçta rastgele seçilmektedir. Dolayısıyla her seferinde farklı sonuçlar verebilir. Burumun önüne geçmek için K-means++ adında çok daha verimli çalışan ve aynı veri için her seferinde aynı sonuçları üreten bir geliştirilmiş hali bulunmaktadır ve günümüzde en yaygın kullanım şekillerinden birisidir[21]. Algoritmanın uygulanması aşamasında gerekli olan bir bilgi de kaç farklı K değeri yani kaç küme olacağının bulunması, en anlamlı sonucu bulmada büyük bir öneme sahiptir. K değerinin kaç olduğunu bulmak için kullanılan en yaygın yöntem elbow metodudur[22].

III. VERİ

Endüstri 4.0 ve beraberinde getirdikleriyle artık fabrikalarda daha akıllı sistemler kullanılmaya başlamıştır. Bu tarz ortamlarda kullanılan endüstriyel robotlar, robot kolları, otonom taşıyıcı araçlar bu akıllı sistemlerin bir parçasıdır. Bu çalışmada da kullanılan veri seti, bir otonom taşıyıcı araç tarafından üretilen titreşim akım verileridir. Her iki veri de araca monte edilen sensörlerden anlık olarak toplanmaktadır. Akım verisi 1 veri / saniye olacak şekilde üretilmektedir. Bunun yanı sıra titreşim verisi zaman serisi bir veridir ve 1200 veri / saniye olacak şekilde üretilmektedir. Veri setine anomali eklemek amacıyla yol üzerine yerleştirilen çeşitli engeller bulunmaktadır. Bu engeller otonom taşıyıcı aracın üzerinden geçebileceği şekilde yerleştirilmiştir. Bu geçişler sırasında titreşim ve akım verilerinde pik noktaların olması sağlanmakta ve anomali olarak değerlendirilmektedir. Kümeleme yaparken bu pik noktaların yakalanması hedeflenmektedir.

IV. YÖNTEM

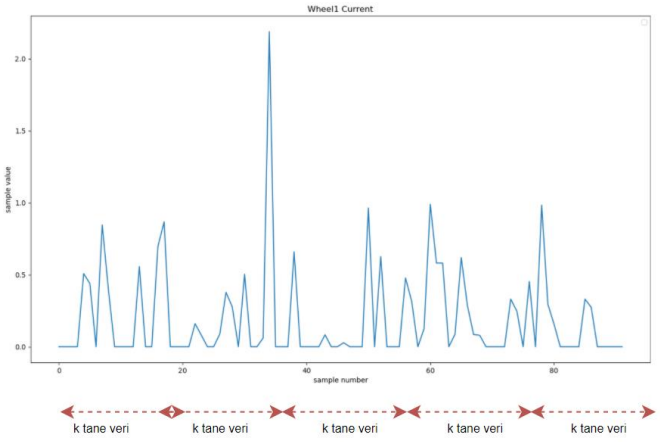
Bu çalışmada iki farklı kümeleme yöntemi kullanılarak anomali tespiti yapılmıştır. Bunlar hiyerarşik kümeleme ve k-means kümelemedir. OTA'nın sensörlerinden toplanan akım ve titreşim verileri bu iki kümeleme algoritması kullanılarak kümelenmiş ve anomaliler tespit edilmiştir. Kümeleme aşamasından önce veriler üzerinde bazı ön işlemler yapılmıştır. Aşağıdaki kullanılan yöntemler verilmiştir.

A. Veri Ön İşlemleri

Akım ve titreşim verileri karakter olarak farklıdır. Akım verisi 10 veri / saniye olacak şekilde üretilmektedir. Titreşim verisi zaman serisi tipindedir ve 1200 veri / saniye olacak şekilde üretilmektedir. Kümeleme yapmadan önce bu veriler üzerinde aşağıda belirtilen işlemler yapılmaktadır.

i. Verileri Gruplara Ayırma

Veriler, kümeleme algoritmasına sokulmadan önce belirli bir oranda(k) gruplanmaktadır. Bu oran paket paket sayısı olarak belirlenir. Örnek olarak k=3 için; her üç paket veri gruplanarak tek bir paket veri gibi kullanılır. Dolayısıyla veri boyutu üçte birine düşer. Bu sayede her bir veri paketi, karakteristiğini daha net göstermektedir. Şekil 1'de akım verisi üzerinde örnek bir gruplandırma gösterilmiştir.



Şekil 1 Veri Gruplandırma (k=gruplardaki veri paketi sayısı)

ii. Özellik Çıkarma

Veriler gruplandıktan sonra artık bir paket içerisinde birden çok veri olacaktır. Aşağıda verilen özellikler bu paketlere ayrı ayrı uygulanmaktadır. Tablo 1'de her özellik için uygulanan yöntem verilmiştir.

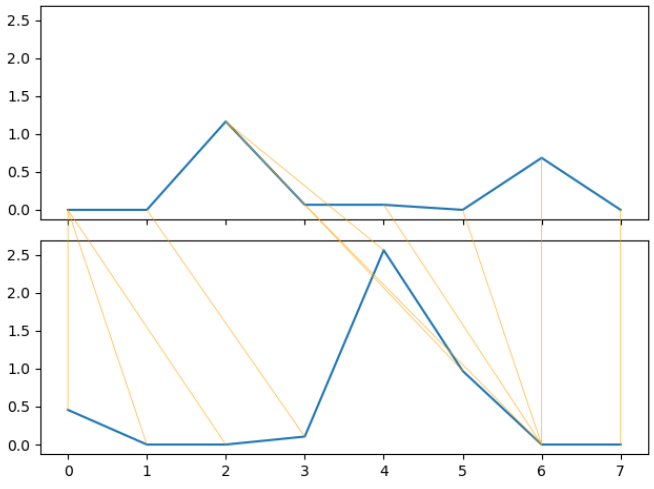
Tablo 1 Veri paketlerinden çıkarılan özellikler

Mean $(\mu) = \frac{1}{N}\sum_{i-1}^{N} X_i$
Median = $((\frac{N+1}{2})^{th})value$
Standard Deviation $(\sigma) = \sqrt{\frac{\sum_{i=1}^{N}(X_i-\mu)^2}{N-1}}$
Variance $(\sigma^2) = \frac{\sum_{i=1}^{N}(X_i-\mu)^2}{N-1}$
Sum = $\sum_{i-1}^{N} X_i$
Skewness $= \frac{1}{N}\frac{\sum_{i=1}^{N}(X_i-\mu)^3}{\sigma^3}$
Kurtosis $= \frac{1}{N}\frac{\sum_{i=1}^{N}(X_i-\mu)^4}{\sigma^4}$
Energy = $\sum_{i=1}^{N} {X_i}^2$
RMS = $\sqrt{\frac{\sum_{i=1}^{N} {X_i}^2}{N}}$

Her veri paketinden bu özellikler çıkarılmaktadır. Özellikler çıkarıldıktan sonra sürece verinin kendini ile değil bu özelliklere göre devam edilmektedir. Yani, her veri paketi, içinde veriye ait 9 özelliği de barındıran bir liste ile temsil edilir.

iii. Dinamik Zaman Atlama (Dynamic Time Warping)

Dinamik zaman atlama, Öklid mesafesi yerine zaman serisi verilerde benzerliği hesaplamak için kullanılır. Ayrıca, zaman serisi analizinde, hız açısından değişebilen iki zamansal dizi arasındaki benzerliği ölçmek için kullanılan algoritmalardan biridir[23]. Şekil 2'de iki zaman serisi verisi arasındaki benzerlik mesafeleri dinamik zaman atlama yöntemi kullanılarak gösterilmiştir.



Şekil 2 Dinamik zaman atlama ile iki zaman serisi verinin mesafelerinin bulunması

iv. Normalizasyon

Verilerden çıkarılan özellikler üzerinde min-max normalizasyon uygulanmaktadır. Normalizasyon uygularken sınır olarak 0 ve 1 kabul edilmiştir. Bu sayede veriler arasındaki olası aşırı fark 0-1 arasına çekilmiştir ve herhangi bir özelliğin sonuca direkt etki etmesi engellenmiştir. Aşağıda min-max normalizasyon formülü verilmiştir.

$$X_{toplam} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

B. Hiyerarşik Kümeleme

Hiyerarşik kümeleme, veri analizi amacıyla kullanılan birçok kümeleme tekniklerinden biridir[6]. Hiyerarşik kümeleme yönteminde başlangıçta tüm veri noktaları ayrı bir küme olarak kabul edilir. İlk olarak birbirine yakın olan iki kümeyi tanımlanır. Daha sonra bu kümeleri birleştirilerek büyük bir küme oluşturulur. Bu iki adım tüm veri seti tek bir kümede toplanana kadar tekrarlanır[24]. Hiyerarşik kümeleme iki kategoriye ayrılabilir: aşağıdan yukarıya(agglomerative) ve yukarıdan aşağıya(divisive)[25]. Yukarıya doğru olan yaklaşımda veriler ayrı bir küme gibi düşünülür. Kümelemeye kendisinden başlar ve etrafındaki diğer kümeler ile birleşerek tüm veri setini çiftler halinde kümeler. Alt kümeleri birleştirirken kullanılan farklı yöntemler vardır. Bunlar; tek (single), tam(complete), ortalama(average), merkez(centroid), medyan(median) ve ward yöntemleridir[26].

G ve H birer küme olmak üzere 'tek' yönteminde bu iki kümeye ait üyelerin herhangi ikisi arasındaki en kısa mesafeye bakılır[27].

$$d_{tek}(G,H) = min_{i\in G, j\in H} d_{ji}$$

G ve H birer küme olmak üzere 'tam' yönteminde bu iki kümeye ait üyelerin herhangi ikisi arasındaki en uzak mesafeye bakılır[27].

$$d_{tam}(G,H) = max_{i\in G, j\in H} d_{ji}$$

G ve H birer küme olmak üzere 'ortalama' yönteminde bu iki kümeye ait üyelerin tüm üyelerin birbirleri ile arasındaki mesafenin toplamına bakılır[27].

$$d_{ortalama}(G,H) = \frac{1}{n_G . n_H}\sum_{i\in G, j\in H} dij$$

Bu yöntemlerin özellikleri [28]'te açıkça anlatılmıştır. Diğer yöntemlerde de bunlara benzer yaklaşımlar kullanılmıştır[29].

C. K-means Kümeleme

K-means algoritması bir gözetimsiz öğrenme ve kümeleme algoritmasıdır. Diğer kümeleme algoritmalarında olduğu gibi bir veri setinde benzer özellikler gösteren verilerin, gruplara ayrılması sağlamaktadır. Bu oluşturulan kümelerin kendi içindeki veriler arasındaki benzerlikler fazla, kümeler arası verilerin benzerlikleri ise azdır.

Temel çalışma yöntemi, veri setinde verilen verilerin sayısal değerlerini uzayda bir koordinat olarak kabul eder ve algoritmanın kabul ettiği K noktasına olan uzaklıkları göz önünde tutularak verinin hangi kümeye dahil olması gerektiğini hesaplar.

Kullanılan veri seti için kaç farklı küme yani K değeri olması gerektiği bilinmiyorsa bu K değerini bulmak için de çeşitli yöntemler kullanılabilir. Bu yöntemlerden birisi olan Elbow (dirsek) metodu, verilen bir aralıktaki K değerlerinin hepsi için K-means algoritmasını çalıştırır ve WCSS (Within Cluster Sum of Square) değerlerini bir grafik şeklinde çizdirir. Grafik genelde çok sert bir kırılmanın olduğu bir nokta barındırdığı ve bir dirseğe benzediği için bu yöntem dirsek (Elbow) metodu olarak adlandırılmaktadır. Tam sert kırılmanın gerçekleştiği nokta optimum K değeridir.

Bu çalışmada K-means algoritması uygulanacak olan veriye Veri başlığında da bahsedildiği gibi, bir ön işleme işlemine tabi tutarak özellik çıkartma ve normalizasyon işlemi uygulanmaktadır.

D. Anomali Skoru Hesaplama

Kümeleme sonuçlarına bakılarak her veri seti için anomali skoru çıkartılır. Bu skorlar verilerin ait oldukları kümenin tüm set içerisindeki sıklığına bağlı olarak değişmektedir. Aşağıda bir C kümesinin anomali skorunun nasıl hesapladığı gösterilmiştir.

$$C_{anomali} = \frac{\#Tüm\ veri\ paketi\ sayısı - \#C\ kümesine\ ait\ veri\ paketi\ sayısı}{\#Tüm\ veri\ paketi\ sayısı}$$

Anomali skoru hem özellik çıkarma hem de dinamik zaman atlama yöntemleri kullanılarak kümelenmiş veriler üzerine ayrı ayrı uygulanmaktadır. Daha sonra bu iki yöntemin anomali sonuçları birleştirilerek yeni bir anomali skoru elde edilir. Skorlar birleştirilirken ortalamaları alınarak kullanılır. Sonuç olarak bir veri paketine ait anomali skoru iki farklı yöntemden gelen ve ortalamaları alınan tek bir skor üretilir. Aşağıda, C kümesine ait özellik çıkarma ve dinamik zaman atlama yöntemleri kullanılarak yapılan kümeleme sonuçlarından elde edilen anomali skorlarının birleştirilmesi gösterilmiştir.

$$C_{ortalama\ anomali\ skor} = \frac{C_{özellik\ çıkarma} + C_{dinamik\ zaman\ atlama}}{2}$$

Anomali skor 0 ve 1 arasında değer almaktadır. Skor 1'e ne kadar yakınsa o veri paketinin anomali olma olasılığı o kadar fazladır.

V. DENEYLER

Deney verileri, OTA'nın hareketleri esnasında üretilmiştir. Bir deney verisi OTA'nın kare hareketi boyunca ürettiği verileri kapsamaktadır ve yaklaşık 60-80 saniye arası sürmektedir. Toplam 6 deney verisi üretilmiştir ve sonuçlar bu veriler üzerinden elde edilmektedir. Deney esnasında hem akım hem de titreşim verileri toplanmıştır. Bu veriler üzerinde k-means ve hiyerarşik kümeleme yöntemleri ile kümeleme yapılmıştır.

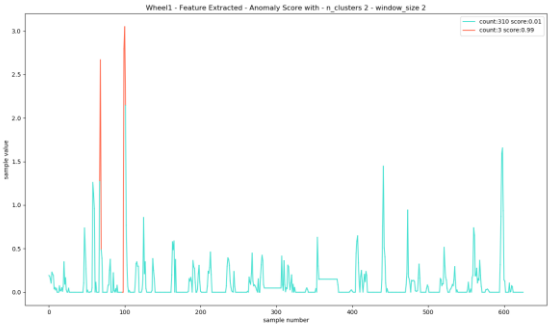
A. Hiyerarşik Kümeleme

Hiyerarşik kümeleme sonuçları dendrogram adı verilen bir şekil üzerinde gösterilmektedir. Aşağıda hem akım hem de titreşim verilerinin çıktıları açıklanmıştır. Çıktılar hem özellik çıkarma hem de dinamik zaman artırma yöntemleri için ayrı ayrı gösterilmiştir.

Akım verisi ile ilgili çıktılar aşağıdaki gibidir.

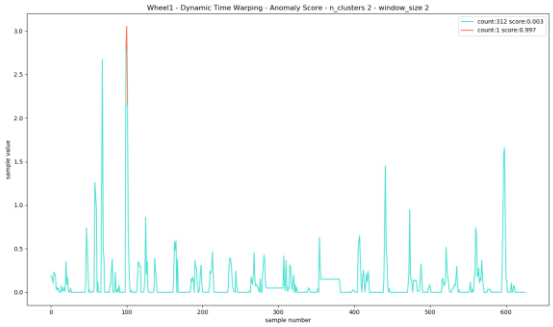
Şekil 3 Akım verisinin dendrogram çıktısı

Şekil 3'te özellik çıkarma yöntemi kullanılarak elde edilen sonuçlar gösterilmiştir. Görüldüğü gibi akım verisinin kümeleme sonucunda sol üst tarafta mavi ile gösterilen ve diğer verilerden daha uzakta duran bir veri bulunmaktadır. Bu veri seti için mavi ile gösterilen veri paketi anomali olarak değerlendirilecektir. Bu paketin gerçek veri üzerinde dağılımı Şekil 4'teki gibidir.



Şekil 4 Akım verisinin özellik çıkarma yöntemi ile hiyerarşik kümeleme sonucunun gerçek veri üzerinde dağılımı

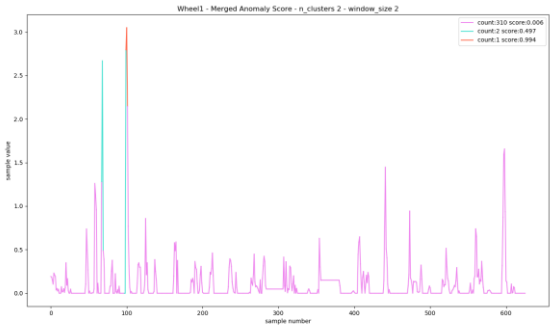
Şekil 4'te özellik çıkarma yöntemi ile yapılan kümeleme sonucu gösterilmektedir. Turuncu ile gösterilen yerlerin diğer verilere oranda daha yukarda olması-pik yapması- onun bir anomali olma olasılığını arttırır. Kümeleme sonucunda da en yüksek iki pik noktası başarılı şekilde yakalanmıştır. Şeklin sağ üst noktasında anomali skorlar ve kümeye ait veri sayısı belirtilmiştir.



Şekil 5 Akım verisinin dinamik zaman atlama yöntemi ile hiyerarşik kümeleme sonucunun gerçek veri üzerinde dağılımı

Aynı verinin bu sefer dinamik zaman atlama yöntemi ile elde edilen sonuçları da Şekil 5'te verilmiştir. Benzer şekilde en yüksek noktanın(turuncu renkli) farklı bir kümeye dahil edilmiştir. Bu yöntemde sadece bir pik noktası anomali olarak kümelenmiştir.

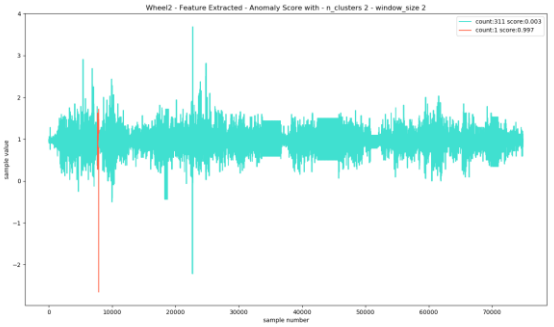
Her iki yönteme de bakıldığında özellik çıkarma yönteminde en yüksek iki pik noktasının da yakalandığı fakat dinamik zaman atlama yönteminde sadece bir pik noktasının yakalandığı görülür. Burada, iki yöntemden elde edilen anomali skorları birleştirildiğinde Şekil 6'daki gibi bir dağılım meydana gelir. Anomali skorlarının birleştirilmesi ile artık üç farklı küme mevcuttur ve en yüksek pik noktasının anomali olma olasılığı en fazladır. İkinci büyük pik noktası da yüksek bir anomali skoruna sahiptir.



Şekil 6 Akım verisinin anomali skorlarının birleştirilmesi ile elde edilen sonuçlar

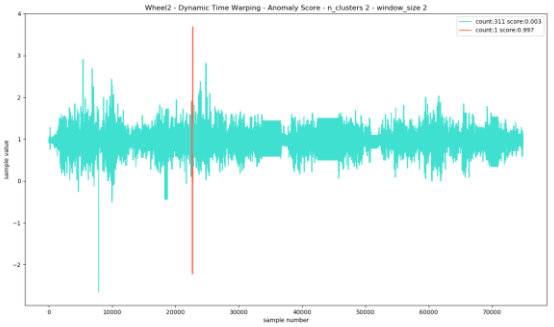
Titreşim verisi ile ilgili çıktılar aşağıdaki gibidir.

Titreşim verisi de her iki yöntem kullanılarak kümelenmiştir. Aşağıda Şekil 7'de özellik çıkarma yöntemi ile elde edilen kümeleme sonucu verilmiştir.



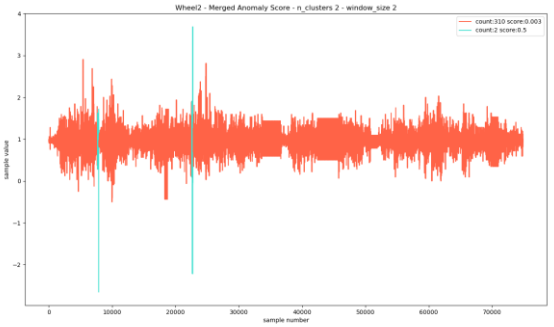
Şekil 7 Titreşim verisinin özellik çıkarma yöntemi ile hiyerarşik kümeleme sonucunun gerçek veri üzerinde dağılımı

Şekil 8'e baktığımızda çok net iki pik noktasından bir tanesinin anomali olarak kümelendiğini ve turuncu renkle gösterildiğini görmekteyiz.



Şekil 8 Titreşim verisinin dinamik zaman atlama yöntemi ile hiyerarşik kümeleme sonucunun gerçek veri üzerinde dağılımı

Diğer taraftan dinamik zaman atlama yönteminden alınan sonuçlara göre iki pik noktasından bir tanesi anomali olarak kümelendirilmiştir. Bu iki çıktıya baktığımızda farkı iki pik noktasını yakaladıkları görülmektedir. Anomali skorları birleştirildiğinde ise aşağıdaki gibi bir çıktı karşımıza çıkmaktadır.



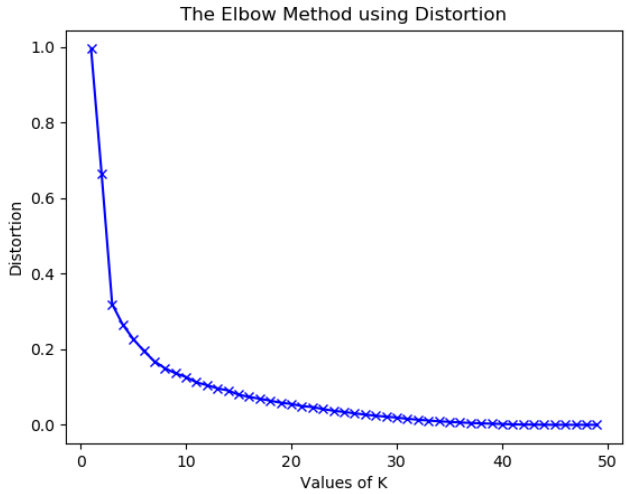
Şekil 9 Titreşim verisinin anomali skorlarının birleştirilmesi ile elde edilen sonuçlar

Şekil 9'e baktığımızda iki yöntemde farklı olarak kümelenen pik noktalar, anomali skorları birleştirildiğinde her ikisinin de anomali olarak değerlendirildiğini görmekteyiz. Bu sayede tek bir yöntem kullanırken göz ardı edilen diğer pik nokta da

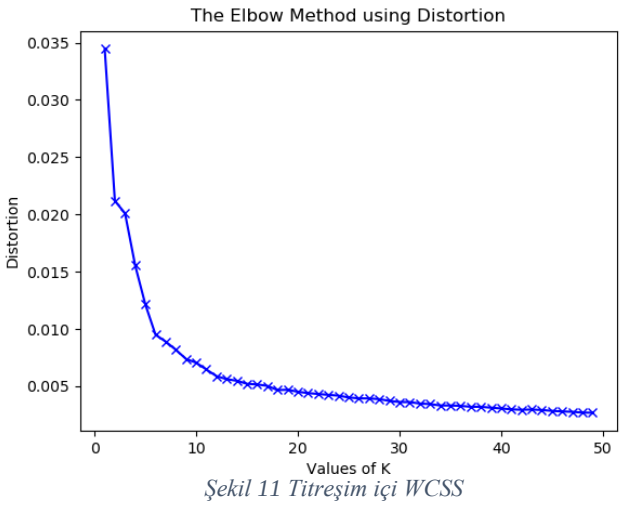
yakalamış ve sonuç olarak en aykırı iki değer anomali olarak bulunmuştur.

B. K-means

K-means kümeleme algoritmasının girdilerinden birisi olan ve doğru bir kümeleme yapabilmek için bulunması gereken öncelikli eleman K değeridir. K değerini yani kaç farklı küme olacağını bulmak için bu veri setinde dirsek (elbow) metodu kullanılmıştır. Şekil 10'da akım verileri için ve Şekil 11'de ise titreşim verileri için hesaplanan WCSS (Within Cluster Sum of Square) değerleri verilmiştir. WCSS değerlerine bakarak grafik üzerindeki kırılma noktasını aldığımızda yani dirsek (elbow) metodu uygulandığında akım verileri için K değeri sekiz ve titreşim için K değeri dokuz olarak alınmıştır.

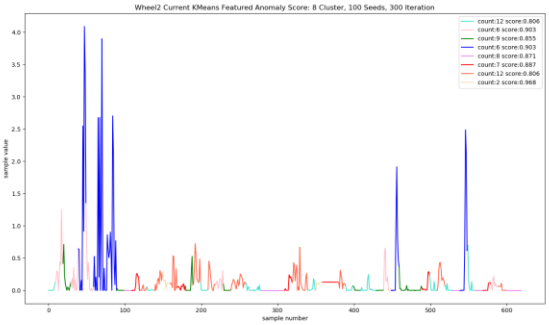


Şekil 10 Akım için WCSS



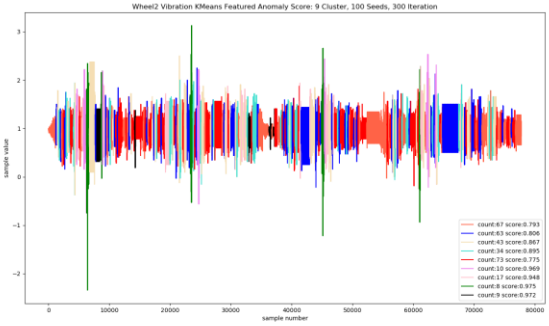
Şekil 11 Titreşim içi WCSS

Akım için bulunan sekiz K değerine göre K-means algoritması akım verisine uygulandığı zaman şekil 10'daki grafik ortaya çıkmaktadır. K-means kümeleme algoritmasını gerçek veri üzerinde gösterebilmek adına grafik üzerinde bulunan aynı renkteki kısımlar bir kümeye aittir. Şekil 12'de göründüğü gibi akım değerlerinde aşırı değişikliklerin veya yüksek değerlere çıkılan gruplar aynı renktedir. Yine aynı şekilde elektrik motorun her zaman akım çekmediği yani akım değerinin sıfır olduğu noktalar ise yine bir başka küme içerisindedir. Her ne kadar bu çalışma için amaç burada anomali durumunu tespit etmek olsa da veri kendi içinde birden fazla karakteristik yapı göstermektedir ve bu durum K değerini arttırmaktadır. Akım değerinin sıfır olduğu durumlar örnekleme olarak az sayıda bulunduğu için bazı durumlarda anomali olarak değerlendirilebilir. Bu örnekte de akımın sıfır olduğu kısımların anomali puanları oldukça yüksek çıkmaktadır.



Şekil 12 Akım değeri üzerinde K-means kümeleme

Titreşim için dokuz olarak bulduğumuz K değeriyle K-means kümeleme algoritması titreşim verisi için uyguladığımızda Şekil 13'deki sonuçlar çıkmaktadır. Grafik üzerinde yeşil renkle gösterilen kısım örneklemenin en az olduğu ve bizim anomali olarak aradığımız kısımdır. Yeşil renkteki grup muhtemel yol üzerinde bulunan bir bozukluktan dolayı ortaya çıkmış bir veridir. Titreşim verisine genel olarak bakıldığında muhtemelen kullanılan sensörden kaynaklanan titreşim değerinin sabitlendiği kısımlar oluşmaktadır. Bu kısımlardaki örneklemenin de fazla olması sebebiyle genelde o kısımdaki veri bloğunun tamamı tek bir kümeye dahil olmaktadır. Akım verisinde olduğu gibi titreşim verisinde titreşimin çok az olduğu kısımlar bulunmaktadır. Örneklem olarak az sayıda bulundukları için anomali skorları yüksek çıkmaktadır.



Şekil 13 Titreşim verisi üzerinde K-means kümeleme

VI. SONUÇLAR

Bu çalışmada laboratuvar ortamında otonom taşıyıcı araç tarafından üretilen akım ve titreşim verileri üzerinde K-means Kümeleme ve Hiyerarşik Kümeleme algoritmalarının kullanılmasıyla elde edilen sonuçlar incelenmiştir. Her iki algoritmanın amacı da veri üzerindeki anomali değerleri bulmaktır. Bu veri formatı için Hiyerarşik Kümeleme ve K-

means Kümeleme algoritmalarının çeşitli avantajları veya dezavantajları bulunmaktadır.

K-means kümeleme algoritması için bahsedecek olursak algoritmanın anomali olduğu ve olmadığı durumlarını sınıflandırırken gerekli olan K değeri değişiklik göstermektedir. Bu durum K-means kümeleme algoritmasıyla bu deneydeki verilerle çalışırken büyük dezavantaj oluşturmaktadır. Bazı farklı veri setlerinde grafikteki değerlerin neredeyse aynı olduğu fakat farklı kümelere dahil olmuş kısımlar da gözlenebilmektedir. K-means kümeleme algoritmasıyla yapılan bir çalışmaya pek rastlanmamaktadır. K-means algoritması, bu deneyde üretilen veri seti veya benzeri veri setleri için kullanılması pek elverişli bir algoritma değildir. Hiyerarşik kümeleme anomalilere karşı dirençli bir algoritma olduğu için anomalileri kolayca tespit edebilmiştir. Kümele sonuçlarına göre tespit edilen anomalilere skor ataması yapılmıştır. Bu skor ne kadar yüksek ise verinin anomali olma olasılığı o kadar yüksek demektir. Anomali skorları özellik çıkarma ve dinamik zaman atlama yöntemleri kullanılarak yapılan kümele sonuçlarından ayrı ayrı çıkartılmıştır. Bu iki yöntemden çıkan skorlar daha sonrasından her veri paketi için ayrı ayrı ortalaması alınarak toplam bir anomali skor ortaya çıkarılmıştır. Bu sayede iki yöntem ile anomaliler doğrulanmış ve skorları ona göre verişmiştir. Bir yöntemin bulamadığı veya daha az skor verdiği anomaliler, toplam skor hesaplandığında gün yüzüne çıkmıştır.

REFERENCES

[1] Borgi, T., Hidri, A., Neef, B., & Naceur, M. S. (2017). Data analytics for predictive maintenance of industrial robots. *International conference on advanced systems and electric technologies (IC_ASET) 2017* (pp. 412–417). IEEE.

[2] Peres, R. S., Dionisio Rocha, A., Leitao, P., & Barata, J. (2018). IDARTS - Towards intelligent data analysis and real-time supervision for industry 4.0. *Computers in Industry*, *101*, 138–146.

[3] Sezer, E., Romero, D., Guedea, F., MacChi, M., & Emmanouilidis, C. (2018). An industry 4.0-enabled low cost predictive maintenance approach for SMEs: a use case applied to a cnc turning centre. *IEEE International conference on engineering, technology and innovation (ICE/ITMC)* (pp. 1–8). IEEE.

[4] Biswal, S., & Sabareesh, G. R. (2015). Design and development of a wind turbine test rig for condition monitoring studies. *2015 International conference on industrial instrumentation and control, ICIC 2015* (pp. 891–896). IEEE.

[5] F. Musumeci et al., "An Overview on Application of Machine Learning Techniques in Optical Networks," in IEEE Communications Surveys & Tutorials, vol. 21, no. 2, pp. 1383-1408, Secondquarter 2019, doi: 10.1109/COMST.2018.2880039.

[6] George Seif, The 5 Clustering Algorithms Data Scientists Need to Know (2018) https://towardsdatascience.com/the-5-clustering-algorithms-data-scientists-need-to-know-a36d136ef68

[7] Thyago P. Carvalho, Fabrízzio A. A. M. N. Soares, Roberto Vita, Roberto da P. Francisco, João P. Basto, Symone G. S. Alcalá, A systematic literature review of machine learning methods applied to predictive maintenance, Computers & Industrial Engineering, Volume 137, 2019, 106024, ISSN 0360-8352, https://doi.org/10.1016/j.cie.2019.106024.

[8] Ariyaluran Habeeb, Riyaz Ahamed & Nasaruddin, Fariza & Gani, Abdullah & Hashem, Ibrahim & Ahmed, Ejaz & Imran, Muhammad. (2018). Real-time big data processing for anomaly detection: A Survey. International Journal of Information Management. 45. 10.1016/j.ijinfomgt.2018.08.006.

[9] Benvenuto, Federico & Piana, Michele & Campi, Cristina & Massone, Anna. (2017). A Hybrid Supervised/Unsupervised Machine Learning Approach to Solar Flare Prediction. The Astrophysical Journal. 853. 10.3847/1538-4357/aaa23c.

[10] Goldstein M, Uchida S (2016) A Comparative Evaluation of Unsupervised Anomaly Detection Algorithms for Multivariate Data. PLoS ONE 11(4): e0152173. https://doi.org/10.1371/journal.pone.0152173

[11] Mohd Zamry, Nurfazrina & Zainal, A. & Rassam, Murad. (2018). Unsupervised anomaly detection for unlabelled Wireless Sensor Networks Data. International Journal of Advances in Soft Computing and its Applications. 10. 172-191.

[12] H. E. Egilmez and A. Ortega, "Spectral anomaly detection using graph-based filtering for wireless sensor networks," *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, 2014, pp. 1085-1089, doi: 10.1109/ICASSP.2014.6853764.

[13] Seetharaman, Sreedhar Kumar. (2016). An Improved Agglomerative Clustering Algorithm for Outlier Detection. Applied Mathematics & Information Sciences. 10. 1141-1154. 10.18576/amis/100332.

[14] Chan, Philip & Mahoney, Matthew. (2005). Modeling Multiple Time Series for Anomaly Detection. 90-97. 10.1109/ICDM.2005.101.

[15] Mingyan Teng, "Anomaly detection on time series," *2010 IEEE International Conference on Progress in Informatics and Computing*, Shanghai, 2010, pp. 603-608, doi: 10.1109/PIC.2010.5687485.

[16] Zhouyu Fu, Weiming Hu and Tieniu Tan, "Similarity based vehicle trajectory clustering and anomaly detection," *IEEE International Conference on Image Processing 2005*, Genova, 2005, pp. II-602, doi: 10.1109/ICIP.2005.1530127.

[17] Dutta N., Kaliannan P., Subramaniam U. (2021) Application of Machine Learning Algorithm for Anomaly Detection for Industrial Pumps. In: Das S., Das S., Dey N., Hassanien AE. (eds) Machine Learning Algorithms for Industrial Applications. Studies in Computational Intelligence, vol 907. Springer, Cham. https://doi.org/10.1007/978-3-030-50641-4_14

[18] O. A. Egaji, T. Ekwevugbe and M. Griffiths, "A Data Mining based Approach for Electric Motor Anomaly Detection Applied on Vibration Data," *2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)*, London, United Kingdom, 2020, pp. 330-334, doi: 10.1109/WorldS450073.2020.9210318

[19] Kilian Hendrickx, Wannes Meert, Yves Mollet, Johan Gyselinck, Bram Cornelis, Konstantinos Gryllias, Jesse Davis, A general anomaly detection framework for fleet-based condition monitoring of machines, Mechanical Systems and Signal Processing, Volume 139, 2020, 106585, ISSN 0888-3270, https://doi.org/10.1016/j.ymssp.2019.106585.

[20] Hartigan, J., & Wong, M. (1979). Algorithm AS 136: A K-Means Clustering Algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, *28*(1), 100-108. doi:10.2307/2346830

[21] Kapoor and A. Singhal, "A comparative study of K-Means, K-Means++ and Fuzzy C-Means clustering algorithms," *2017 3rd International Conference on Computational Intelligence & Communication Technology (CICT)*, Ghaziabad, 2017, pp. 1-6, doi: 10.1109/CIACT.2017.7977272.

[22] D. Marutho, S. Hendra Handaka, E. Wijaya and Muljono, "The Determination of Cluster Number at k-Mean Using Elbow Method and Purity Evaluation on Headline News," *2018 International Seminar on Application for Technology of Information and Communication*, Semarang, 2018, pp. 533-538, doi: 10.1109/ISEMANTIC.2018.8549751.

[23] https://en.wikipedia.org/wiki/Dynamic_time_warping

[24] Shukla, Raj & Sengupta, Shamik. (2020). Scalable and Robust Outlier Detector using Hierarchical Clustering and Long Short-Term Memory

(LSTM) Neural Network for the Internet of Things. Internet of Things. 9. 100167. 10.1016/j.iot.2020.100167.

[25] M Steinbach, G Karypis, and V Kumar. A Comparison of Document Clustering Techniques. *KDD Workshop on Text Mining*, 400(X):1–2, 2000

[26] Thuy-Diem Nguyen, Bertil Schmidt, Chee-Keong Kwoh, SparseHC: A Memory-efficient Online Hierarchical Clustering Algorithm, Procedia Computer Science, Volume 29, 2014, Pages 8-19, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2014.05.001.

[27] Pierpaolo D'Urso, Vincenzina Vitale, A robust hierarchical clustering for georeferenced data, Spatial Statistics,Volume 35, 2020, 100407, ISSN 2211-6753, https://doi.org/10.1016/j.spasta.2020.100407.

[28] Brian S Everitt, Sabine Landau, Morven Leese, Daniel Stahl, Walter A Shewhart, and Samuel S Wilks. Cluster Analysis, 5th Edition. Wiley Series in Probability and Statistics. 2011

[29] G. N. Lance and W. T. Williams. A General Theory of Classificatory Sorting Strategies: 1. Hierarchical Systems. The Computer Journal, 9(4):373–380, February 1967