



## Estimation of clustering parameters and anomaly detection in tracking devices with changeable position time

Mustafa Tolga Datlıca<sup>1</sup> , Erman Çakıt<sup>2\*</sup> 

<sup>1</sup>Atel Teknoloji ve Savunma A.Ş., Ankara, Turkey

<sup>2</sup>Department of Industrial Engineering, Gazi University, Ankara, 06570, Turkey

### Highlights:

- Anomalies in the location behavior
- A density-based clustering algorithm
- Estimation of clustering parameters

### Keywords:

- Tracking devices
- Clustering algorithms
- ST-DBSCAN
- Machine learning
- Anomaly detection

### Article Info:

Research Article  
Received: 31.12.2019  
Accepted: 23.08.2020

### DOI:

10.17341/gazimmfd.668215

### Correspondence:

Author: Erman Çakıt  
e-mail: ecakit@gazi.edu.tr  
phone: +90 312 582 3820

### Graphical/Tabular Abstract

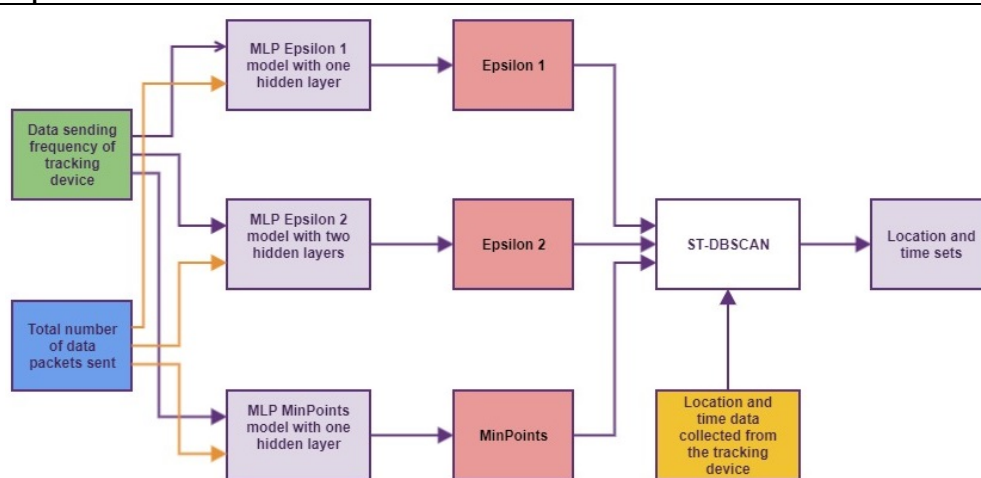


Figure A. Overview of applied methods

**Purpose:** In this study, it was aimed to detect anomalies in the location behavior of objects followed by a tracking device.

### Theory and Methods:

ST-DBSCAN (Spatial-Temporal Density-Based Spatial Clustering of Applications with Noise) density-based clustering algorithm was applied on the data obtained, and weekly patterns were determined for the subject to be located at which time intervals. The input parameters of the ST-DBSCAN algorithm vary according to the frequency of the data from the tracker and the total number of data packets. In this context, the parameters used in the St-DBSCAN algorithm, as well as the frequency of sending data and the number of data packets, are labeled according to the behavior of the object being followed. On these tagged data, linear regression and artificial neural networks methods were compared and a model was proposed that could predict clustering parameters.

### Results:

Weekly patterns were determined by methods developed using information about the object being followed, and these patterns were considered to be normal behaviors of the object being tracked. The instantaneous position is defined as an anomaly if the data obtained is contrary to the pattern.

### Conclusion:

In this study, the temporal data of the object, called time and coordinate data, were grouped with similar ones with the help of clustering algorithms, and were defined as anomalies when data other than sets that were normally considered to be able to predict with these sets.



## Değiştirilebilir konum süresine sahip takip cihazlarında kümeleme parametrelerinin tahmini ve anormallik tespiti

Mustafa Tolga Datlıca<sup>1</sup> , Erman Çakıt<sup>2\*</sup> 

<sup>1</sup>Atel Teknoloji ve Savunma A.Ş., Ankara, Türkiye

<sup>2</sup>Gazi Üniversitesi, Mühendislik Fakültesi, Endüstri Mühendisliği Bölümü, 06570 Maltepe Ankara, Türkiye

### Ö N E Ç İ K A N L A R

- Konum davranışlarındaki anormallikler
- Yoğunluk bazlı kümeleme algoritması
- Kümeleme parametrelerinin tahminini

#### Makale Bilgileri

Araştırma Makalesi  
Geliş: 31.12.2019  
Kabul: 23.08.2020

#### DOI:

10.17341/gazimmfd.668215

#### Anahtar Kelimeler:

İzleme cihazları,  
kümeleme algoritmaları,  
ST-DBSCAN,  
makine öğrenmesi,  
anormallik tespiti

#### ÖZET

Bu çalışmada, takip cihazı kullanılarak konum ve zaman verilerinin toplanması ve takip edilen nesnelerin konum davranışlarındaki anormalliklerin tespit edilmesi amaçlanmıştır. Elde edilen verilere ST-DBSCAN (Spatial-Temporal Density-Based Spatial Clustering of Applications with Noise) yoğunluk bazlı kümeleme algoritması uygulanmış ve takip edilen nesneye ait hangi zaman aralıklarında nerede olduğuna dair haftalık örüntüler tespit edilmiştir. ST-DBSCAN algoritmasının girdi parametreleri, takip cihazından gelen verinin sıklığı ve toplam veri paketi sayısına göre değişiklik göstermiştir. Takip edilen nesnenin davranışlarına göre ST-DBSCAN algoritmasındaki parametreler kullanılarak veri gönderme sıklığı ve veri paketi sayısı etiketlenmiştir. Etiketlenen veriler kullanılarak doğrusal regresyon ve yapay sinir ağları yöntemleri karşılaştırılmış ve kümeleme parametrelerinin tahminini yapabilecek bir model önerilmiştir. Haftalık örüntüler, geliştirilen yöntemler ile takip edilen nesneye ait bilgiler kullanılarak belirlenmiş ve bu örüntüler takip edilen nesneye ait normal davranışlar olarak kabul edilmiştir. Anlık konumu elde edilen veri örüntüye aykırı ise anormal olarak tanımlanmıştır. Böylece normal davranışı bilinen nesnenin, normal davranış desenine uymayan davranışları karşılaştırılarak anormallik tespitini yapabilecek bir yöntem önerilmiştir. Önerilen yöntem farklı gruplar için (çocuk, yaşlı, hasta vb.) erken uyarı sistemi olarak kullanılabilir.

## Estimation of clustering parameters and anomaly detection in tracking devices with changeable position time

### H I G H L I G H T S

- Anomalies in the location behavior
- A density-based clustering algorithm
- Estimation of clustering parameters

#### Article Info

Research Article  
Received: 31.12.2019  
Accepted: 23.08.2020

#### DOI:

10.17341/gazimmfd.668215

#### Keywords:

Tracking devices,  
clustering algorithms,  
ST-DBSCAN,  
machine learning,  
anomaly detection

#### ABSTRACT

In this study, it was aimed to detect anomalies in the location behavior of objects followed by a tracking device. ST-DBSCAN (Spatial-Temporal Density-Based Spatial Clustering of Applications with Noise) density-based clustering algorithm was applied on the data obtained, and weekly patterns were determined for the subject to be located at which time intervals. The input parameters of the ST-DBSCAN algorithm vary according to the frequency of the data from the tracker and the total number of data packets. In this context, the parameters used in the St-DBSCAN algorithm, as well as the frequency of sending data and the number of data packets, are labeled according to the behavior of the object being followed. On these tagged data, linear regression and artificial neural networks methods were compared and a model was proposed that could predict clustering parameters. Weekly patterns were determined by methods developed using information about the object being followed, and these patterns were considered to be normal behaviors of the object being tracked. The instantaneous position is defined as an anomaly if the data obtained is contrary to the pattern. Thus, a method has been proposed to detect anomalies by comparing the behavior of the object known to be normal behavior that does not fit the normal behavior pattern. The proposed method can be used as an early warning system for different groups (children, elder people, sick people, etc.).

\*Sorumlu Yazar/Corresponding Author: mdatlica@atel.com.tr, ecakit@gazi.edu.tr / Tel: +90 312 582 3820

## 1. GİRİŞ (INTRODUCTION)

Makine öğrenmesi literatürde büyük veri içeren istenmeyen e-postaların tanınması (Spam Detection), hile denetimi (fraud detection), rakam tanıma (digit recognition), doğal dil işleme (natural language processing), yüz tanıma (face detection), ürün tavsiye etme (product recommendation), medikal teşhis (medical diagnosis), hisse senedi alım satım (stock trading), müşteri bölümlenme (customer segmentation), ve şekil tanıma (shape detection) alanlarında sıklıkla kullanılmaktadır [1].

Takip cihazlarındaki gelişmeler, cihazların artık çok küçük boyutlarda kullanıldığını göstermektedir. Bu teknolojik gelişmeler ile giyilebilir ve taşınabilir cihaz tasarımlarında hızlı bir artış görülmektedir. Konum tabanlı arama gibi çeşitli uygulamalar için elektronik cihaz ve araçların büyük bölümü navigasyon ve (Küresel Konumlandırma Sistemi) GPS modülü ile donatılmıştır [2]. Bütün GPS sistemlerinde temel olarak enlemden ve boylamdan oluşan koordinat diğer bir adı ile uzamsal bilgiler üretilir. Takip sistemlerinde ek olarak zamansal veri de bunlara eklenir. Bu iki türdeki veri uzamsal-zamansal (spatial-temporal) veri olarak adlandırılır. Diğer bir ifade ile uzamsal-zamansal veriler, uzamsal veri kümesinin zamansal dilimleri olarak depolanan verileri belirtir [3]. Mekânsal-zamansal veriler, mekânsal ve zaman boyutlarına göre toplanmakta olup, zaman ve koordinat açısından verilerin yoğunlaştığı yerler küme analizi ile tespit edilebilir [3]. Küme analizi; veri bölümlenme, veri azaltma, aykırı değer tespiti, gürültü filtreleme, örüntü tanıma ve görüntü işleme dahil birçok mühendislik ve bilimsel uygulama alanında sıklıkla kullanılmaktadır [3]. Kümeleme ya da küme analizi, örnekler arasındaki benzerlikleri bulup, benzerleri bir araya getirmek olarak tanımlanabilir [4]. Kümeleme algoritmalarının amacı ise bir veri setini anlamlı gruplara ayırmaktır [5]. Algoritması olmayıp örnek veriye sahip olan birçok uygulamada makine öğrenimi kullanılabilir [4]. Bu kapsamda, takip cihazlarında konum anlık olarak izlenebilmektedir. Ancak sistemlerin sadece bilgi almak için kullanılması yerine, bilgilendirici olması adına takip sistemleri tarafından uyarılar verebilmesi planlanmaktadır. Hâlihazırda var olan bu uyarılar standart parametrelere, kullanıcı girişine ve sabit algoritmalara bağlı olduğu için günlük hayat içerisinde karşılık bulamamaktadır. Ayrıca her takip edilecek nesne için ayrı bir algoritma yazılması pek mümkün olmamaktadır.

Anormallikler, iyi olarak tanımlanmış bir normal davranış durumuna uymayan veri kalıplarıdır [6]. Genel olarak, çoğu anormallik saptama yöntemi normal durumların bir profilini oluşturur, ardından normal profile uymayan anormallikleri tanımlar [7]. Diğer bir ifade ile anormallik tespiti, veri içerisinde beklenen davranışa uymayan desenlerle ilgili problemi belirlemeyi amaçlar. Literatürde farklı anormallik tespit teknikleri geliştirilmiş olup, bu tekniklerin çoğu belirli uygulama alanları için özel olarak geliştirilmiştir [6]. Bu yöntemler mesafeye dayalı, kümeleme temelli, sınıflandırma temelli, ve istatistiksel anormallik tespit yöntemleri olarak

sınıflandırılmıştır [7]. Kümeleme temelli yöntemlerde, normal gözlemlerin aynı kümeye / kümelere ait olduğu varsayılmaktadır. Yeni bir gözlem, küme merkez (ler) inden uzaktaysa, anormallik olarak değerlendirilmektedir [7]. Gelişen yapay zekâ ve makine öğrenimi teknikleri ile GPS verilerinin anormal davranışlar sergileyip sergilemediği tespit edilebilir. Ayrıca, yapay zekâ ve makine öğrenimi ile geçmiş veriler kullanılarak gelecek ile ilgili tahmin modelleri oluşturulabilir [4].

Bu çalışma kapsamında, literatürdeki eksikliklerden yola çıkılarak, takip edilen nesnenin ve yapay sinir ağları kullanılarak belirlenen takip cihazının özelliklerine göre kümeleme algoritma girdileri belirlenmiş, dışarıdan herhangi bir müdahaleye gerek kalmadan rutinleri ortaya çıkarabilecek bir yöntemler dizisi oluşturulmuştur. Böylece normal davranış bilinen nesnenin, normal davranış desenine uymayan davranışları karşılaştırılarak anormallik tespitini yapabilecek bir yöntem önerilmiştir. Önerilen yöntem farklı gruplar için (çocuk, yaşlı, hasta vb.) erken uyarı sistemi olarak kullanılabilir.

## 2. LİTERATÜR ARAŞTIRMASI (LITERATURE RESEARCH)

Kümeleme algoritmaları konum verileri üzerinde sıkça kullanılan algoritmalarındandır. Hong vd. [8] çalışmalarında, toplu taşıma araçları üzerinden elde edilen konum verilerinden seyahat desenleri tespit edilmiştir. Çalışmanın odak noktası çeşitli akıllı ulaşım sistemi uygulamalarını desteklemek için ulaşım ağlarında araç yörüngesi kümelenmesini ve yolculuk düzenini tanımadır. Çalışmada hareket analizi için mekansal-zamansal veri madenciliğinin daha fazla gelişmeye ihtiyaç duyduğu vurgulanmış ve ST-TOPOSCAN adı verilen yoğunluk bazlı bir algoritma önerilmiştir. Bu kapsamda Şikago şehrinde hareket halindeki taşıtlardan 200,000 taşıt rotası ele alınmış, otoyol ve normal yollara ilişkin sabah saatlerinde farklı yoğunluklar tespit edilmiştir. Yapılan araştırma sonucunda sabah saatlerinde özellikle 06:30, 07:30 ve 08:00 saatlerinde yapılan kalkışların, sabah karşılaşılan trafik sıkışıklığına sebep olduğu ortaya konmuştur. Yuan vd. [9] çalışmalarında taksiler üzerinden elde edilen konum verilerinden kümeleme algoritmaları yardımı ile en kısa rotalar elde edilmiştir. Bu kapsamda çok sayıda taksinin geçmiş tarihli GPS verileri alınmış ve belirli bir kalkış zamanında belirtilen hedefe ulaşabilmek için en hızlı yol tahmini yapılmıştır. Ayrıca, taksilere ait dinamik yol ağları, sıkça geçtikleri bir yol doğrultusu üzerinde zamana bağlı olarak modellenmiştir. Ardından iki yer arasındaki seyahat süresini tahmin etmek için varyans-entropi tabanlı kümeleme (VE-Clustering) adı verilen bir kümeleme yöntemi kullanılmıştır. Sonuç olarak, oluşturulan sistem tarafından önerilen rotalar, diğer sistemlere göre yüzde 60 ile yüzde 70 arasında daha hızlı olduğu tespit edilmiştir. Bu çalışma ile konum ve zaman verisi üzerinde uygulanan kümeleme algoritmalarının, rota tahmini gibi zor problemlerde dahi iyi sonuçlar verdiği sonucuna varılmıştır.

Lee ve Whang [10] konum verileri üzerinde kullanılan kümeleme algoritmalarına ilişkin olarak, takip edilen konum verilerinin alt rotalarına odaklanmışlardır. Çalışmalarında kümeleme rotalarının alt rotalarının önemli olduğu senaryolar vurgulanmış, alt rotaların bulunmadığı durumlara TRACCLUS adını verdikleri algoritma ile çözüm önerisinde bulunmuşlardır. Çalışma kapsamında kasırgaların rotalarına ilişkin bir veri seti ve hayvan hareketlerinin bulunduğu bir başka veri seti üzerinde uygulamalar yapılmıştır. Veri setine uygulanan TRACCLUS algoritması ile alt rotaların oldukça doğru şekilde tespit edildiği görülmüş ve konum verileri üzerinde kümeleme algoritmalarının iyi sonuçlar verdiği sonucuna varılmıştır.

Bu çalışma kapsamında zaman, enlem ve boylam verileri üzerine odaklanılan bir başka konu ise rotalama problemidir. Bu problemin çözümünde bu çalışmada kullanılan kümeleme algoritmalarına benzer DBSCAN algoritmasının genişletilmiş ya da özelleştirilmiş uyarlamaları kullanılmaktadır. Won vd. [11] T-DBSCAN adı verilen ve DBSCAN algoritmasının uyarlaması olan bir algoritma geliştirmiştir. Çalışma kapsamında Şanghay şehrinde yer alan 10,000 rota üzerinde çalışılmıştır. Veriler işleme hazırlandıktan sonra FastMap ve hiyerarşik kümeleme algoritmaları uygulanmış olup, elde edilen sonuçlar işinde uzman olan kişilerce belirlenen sonuçlar ile karşılaştırılmıştır. Çalışma sonucunda öneri olarak sunulan çözüm, uzman bir kişi ile elde edilebilecek sonuçlara göre yüzde 95 oranında daha verimli bulunmuştur.

Cep telefonları gibi GPS özelliği bulunan cihazlar sayesinde konum ve zaman verilerinin erişimi artık oldukça kolaydır. Zhang vd. [12] cep telefonlarından toplanan bilgiler ile tıp merkezlerinin ne kadar ziyaret edildiği ve bu ziyaretlere ait desenlerin tespiti üzerine odaklanmıştır. Çalışmada semantik olarak anlamlı vakit geçirilen yerleri tanımlamak için Gizli Markov Modeli kullanılmış olup, zaman ve konum verilerini kümelemek için TRACCLUS kümeleme algoritması kullanılmıştır. Kullanılan gizli markov modeli ile takip edilen nesnelere tıbbi bir tedavi için belirtilen yerde olup olmadığı ayırımı yapılmıştır. Çalışma sonucunda sık sık tedavi merkezleri ziyaret edenler ile tedavi merkezlerini az ziyaret edenler arasında bir ayırım yapılabilmektedir.

Taksi yolculuklarından üretilen küresel konumlandırma sistemi (GPS) verileri, yüksek mekansal-zamansal çözünürlüğe sahip kentsel nüfusların seyahat davranışları hakkında fikir veren değerli bir bilgi kaynağıdır. Buradan elde edilen bilgiler ile günlük aktiviteler hakkında analizler yapılabilir ve şehir planlaması adına çıkarımlar elde edilebilir. Gong vd. [13] iki katmanlı birleşik bir çözüm sunarak bu aktivite rutinleri üzerine odaklanmıştır. Bu kapsamda k-Means algoritması ile iki katmanlı yöntem karşılaştırılmış olup, birden fazla yöntem ile daha başarılı tahminlerin konum ve zaman verileri üzerinden elde edilebileceği sonucuna varılmıştır.

Nesnelerin interneti (IoT) kullanımının artışı ile gerçek zamanlı olarak nesnelere izlenmesi kolaylaşmış ve bu

nesnelere hakkında GPS verilerinin toplanması sonucu üzerinde çalışılabilecek birçok veri elde edilmiştir. Literatürde yapılan birçok çalışmada yoğunluk bazlı kümeleme algoritmalarının kullanımı bu veriler üzerinde yapılmaktadır. Wang vd. [14] büyük yük gemilerinin ve yolcu taşıyan gemilerin kullanılması zorunlu olan otomatik tanımlama sistemi üzerinden elde edilen konum verilerinden yararlanmışlardır. Çalışma kapsamında dinamik zaman bükme (Dynamic Time Warping, DTW) algoritması ile HDBSCAN hiyerarşik yoğunluk bazlı algoritma kullanılmıştır. Sonuç olarak HDBSCAN kümeleme algoritmasının kümeleme algoritmalarının rota ve yoğunluk tahmininde başarılı sonuçlar verdiği görülmüştür.

Akıllı sensörler, akıllı telefonlar ve sosyal medyanın hızlı gelişimi ile dağıtılmış sensörler ve izleme sistemleri aracılığı ile büyük miktarlarda uzamsal veri üretilmektedir. Bu üretilen verilerin coğrafi benzerlikleri çoğu kez kümeleme algoritmaları ile tespit edilirken, coğrafi benzerliklerin sıcaklık, hız, zaman ve birçok semantik benzerlikleri de kümeleme algoritmaları ile ölçülebilir. Wai ve Nwe Aung [15] çalışmalarında enlem ve boylamdan oluşan bir konum verisinin yanında diğer bir üçüncü boyut verisi olarak semantik verilerin kullanılabilirliği ve kümeleme algoritmalarında bu verilere ilişkin bir uzaklık ölçüm yöntemi üzerine odaklanmıştır. Bu çalışmada uygulandığı gibi konum verilerini oluşturan enlem ve boylam bilgisinin yanında farklı bilgiler de kullanılarak, yakınlıklarının ölçülebilirliği ve kümelerin tespit edilebileceği görülmüştür.

Kümeleme algoritmaları, özellikle ST-DBSCAN algoritması, birçok alanda kullanılmakla beraber, salgın hastalıkları konu alan tıp alanı olan epidemiyoloji gibi çok özel alanlarda da uygulanmıştır. Chimway ve Anuradha [16], epidemiyolojide, yeni bilgileri keşfetmek için uzamsal-zamansal kümeleme gibi yöntemlerden faydalanılarak bazı hastalıklara ait örüntülerin ve eğilimlerin tespit edilebileceği vurgulanmıştır. Bu kapsamda, 2007 ile 2017 yılları arasında Batı Nil Virüsü olarak bilinen hastalığa ait zaman ve konum verilerinden yararlanılmıştır. Çalışmada, zamansal niteliklerin uzaklıklarını belirleme konusunda belirsizlik olduğu için, ST-DBSCAN algoritmasının bulanık mantık uyarlaması olan ST-Fuzzy DBSCAN (ST-FDBSCAN) algoritmasını uygulamışlardır. Literatürde yer alan diğer uzam-zamansal veri tipi ile veri madenciliği yaklaşımlarının kullanıldığı çalışmalar Tablo 1'de özetlenmiş ve mevcut çalışma ile karşılaştırılmıştır. Literatürde yer alan çalışmalar, sadece kümeleme algoritmalarını veriler üzerine uygulamıştır. Bu çalışmada ise kümeleme algoritmalarının parametrelerinin tahmini yapılmış ve özgün veriler kullanılmıştır. Özgün ve saha verilerinin kullanılması bu çalışmayı diğer çalışmalara göre daha özgün kılmaktadır. Ek olarak, literatürde çok az yer alan bir algoritma olan ST-DBSCAN kümeleme algoritması bu çalışma kapsamında kullanılmıştır. Diğer çalışmalar sık kullanılan kümeleme algoritmaları üzerine odaklanmaktadır. Yapay sinir ağları da kullanarak diğer çalışmalarda yer almayan farklı hiperparametreler ile sonuçların değerlendirilmesi de bu çalışmayı diğer çalışmalardan ayırmaktadır.

**Tablo 1.** Uzam-zamansal veri tipi ile veri madenciliği metodları kullanılan çalışmalar  
(Studies on spatiotemporal data types and data mining techniques proposed for them)

İlk yazar	Uzam- zamansal veri tipi	Veri madenciliği metodu	Kullanılan algoritma
Chimwayi [16]	Noktalar (olaylar)	Kümeleme	ST-DBSCAN
Trisminingsih [17]	Noktalar (olaylar)	Kümeleme	ST-DBSCAN
Wang [18]	Poligonlar (coğrafi alanlar)	Kümeleme	ST-SNN
Briant [3]	Noktalar (olaylar)	Kümeleme	ST-DBSCAN
Joshi [19]	Poligonlar (coğrafi alanlar)	Kümeleme	STPC
Wang [20]	Noktalar (olaylar)	Kümeleme	ST-DBSCAN
Chen [21]	Noktalar (olaylar)	Kümeleme	ST-DBSCAN
Wang [22]	Poligonlar	Kümeleme	Multilevel graph-partition algorithms
Zhang [23]	Poligonlar (coğrafi alanlar)	Kümeleme	ST-SSN
Damiani [24]	Noktalar / yörüngeler	Kümeleme	SeqScan
Izakian [25]	Noktalar (zaman serileri)	Kümeleme	Fuzzy c-means
Izakian [26]	Noktalar (zaman serileri)	Kümeleme / anormallik tespiti	Fuzzy c-means
Lai [27]	Noktalar (olaylar)	Kümeleme	DBSCAN
Iyengar [28]	Ayrık olaylar	Kümeleme	Spatial scan statistics
Schubert [29]	Ayrık olaylar	Anormallik tespiti	kNN
Mohan [30]	Ayrık olaylar	Desen analizi	CSTPM
Shekhar [31]	Ayrık olaylar	Desen analizi / anormallik tespiti	Spatial statistics
Nanni [32]	Yörüngeler	Kümeleme	OPTICS
Li [33]	Yörüngeler / Ayrık olaylar	Kümeleme / desen analizi	Spatiotemporal pattern mining
Palma [34]	Yörüngeler	Kümeleme	CB-SMOT
Gudmundsson [35]	Hareketli nesnelere	Kümeleme	Geometric algorithms
Jeung [36]	Hareketli nesnelere	Kümeleme	CuTS
Li [37]	Hareketli nesnelere	Kümeleme	SWARM
Mevcut çalışma	Noktalar (olaylar)	Kümeleme / anormallik tespiti	ST-DBSCAN

Çalışmanın takip eden bölümünde materyal ve yöntemle ait gerekli tanımlar ve kullanılan formüller verilmekte ve önerilen yöntemin adımları ve önerilen yöntemin nasıl çözümler getirdiği vurgulanarak açıklanmaktadır. Bölüm 4'te uygulamaya dair bilgiler verilmekte ve son olarak Bölüm 5'te sonuçlar ve öneriler sunulmuştur.

### 3. MATERYAL VE YÖNTEM (MATERIAL AND METHODOLOGY)

Bu çalışma kapsamında özel bir savunma sanayi firması tarafından üretilmiş takip cihazı kullanılarak veriler toplanmıştır. Kullanılan cihaz, takip cihazı olarak, üzerinde bulunan GPS modül aracılığıyla konum gönderen bir nesnelere interneti (IoT) cihazıdır. Standart takip cihazlarına ek olarak GPS uydularından gerekli veri alınmaz ise Wi-Fi ve konum bazlı servisler (LBS) üzerinden konumlandırma yapabilmektedir. Cihazın temel prensibi GPS, Wi-Fi veya LBS bilgilerini elde edip, bir sunucuya iletmektir. Sunucu tarafından alınan veri, konumlandırma için kullanılmakta ve

veritabanına kaydedilmektedir. Bir web arayüz aracılığıyla cihazın topladığı verilerin enlem, boylam ve zaman bilgileri, kullanıcı tarafından alınabilmektedir. Kullanılacak cihaz her ne kadar üzerinde bulunan sensörlerden (sıcaklık, ivme vb.) farklı bilgiler gönderse de bu veriler çalışma kapsamına alınmamıştır. Çalışma kapsamında kullanılacak veri, tüm takip cihazlarında ortak olan uzamsal-zamansal veridir. Diğer takip cihazlarına benzer şekilde bu cihazdan da enlem, boylam ve zaman bilgileri alınabilmektedir. Bir diğer özellik ise cihazın belli zaman aralıklarında göndermekte olduğu verinin, gönderme sıklığı ayarlanabilmektedir. Bu bakımdan cihaz üzerinden farklı zaman aralıklarında veri toplanabilmesine imkan vermektedir.

Çalışma kapsamında kullanılan takip cihazı ile 4 hafta boyunca veriler toplanmıştır. Takip cihazının veri gönderim sıklığı 5 dakika olarak belirlenmiştir. Bu kapsamda cihaz 5 dakikada bir aktif hale gelmekte, cihazın konumuna ilişkin verileri toplamakta, veriyi sunuculara gönderdikten sonra tekrar uyku moduna geçmektedir. Veri toplama işlemi



sırasında, çalışma kapsamında toplanacak verilerin davranışına uygun olmasına dikkat edilmiştir. Takip cihazının belli bir rutin içerisinde olan bir kişi tarafından taşınması, mümkünse veri toplama sırasında tatil günleri, izin gibi dönemlere denk gelmemesi ve bu sayede takip cihazını taşıyan kişiye ilişkin sağlıklı verinin toplanabilmesi hedeflenmiştir. Cihazın taşınabilir olması bu kısımda ek bir avantaj sağlamıştır. Çalışmanın temel girdileri, takip cihazında elde edilecek olan konum (enlem ve boylam) ve zaman bilgisidir. Bunlara ek olarak, sisteme cihazın kaç adet konum paketi biriktirdiğini belirten bir girdi ve dakika cinsinden ne kadar zaman aralığında veri paketleri gönderdiği bilgisi sistemin girdisi olarak ele alınmıştır.

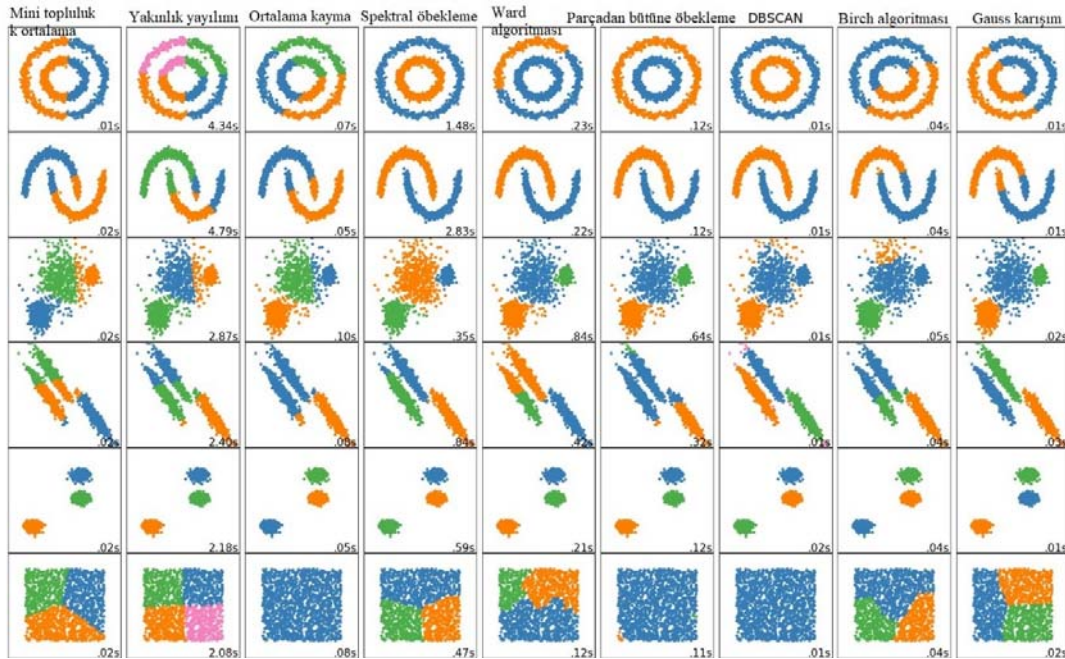
### 3.1. Makine Öğrenmesi (Machine Learning)

Makine öğrenimi ya da yapay öğrenme, bilgisayarların örnek veri ya da geçmiş deneyimi kullanarak bir ölçüte göre başarımlarını artıracak biçimde programlanmasıdır [6]. Takip cihazlarından oldukça fazla veri akışı gerçekleşmekte olup, bu çalışma kapsamında, gözetimli ve gözetimsiz öğrenme tipleri kullanılarak verilerin içerisinde birbirine benzer olanların bir örüntü oluşturup oluşturmadığı tespit edilecektir. Gözetimli öğrenmede eğitim kümesi sınıf adlarını içerir, yani eğitim kümesinde ki her bir nokta için doğru kategoriler verilir [5]. İlk aşamada algoritma eğitim verisini kullanarak tüm kategorileri öğrenir, daha sonra ikinci safhada gelen yeni veri için öğrenilen sınıflardan birini tahmin eder [1]. Gözetimli öğrenmede amaç, girdi değeri ile doğru değeri bir gözetmen tarafından verilen çıktı değeri arasında ki ilişkiyi öğrenmektir. Gözetimsiz öğrenmede amaç girdi değerindeki düzenli durumları bulmaktır [6]. Gözetimsiz öğrenme algoritmasının veri içerisindeki grupları bulması beklenir [6]. Kümeleme en çok kullanılan gözetimsiz öğrenme metodlarından birisi olup, büyük veri

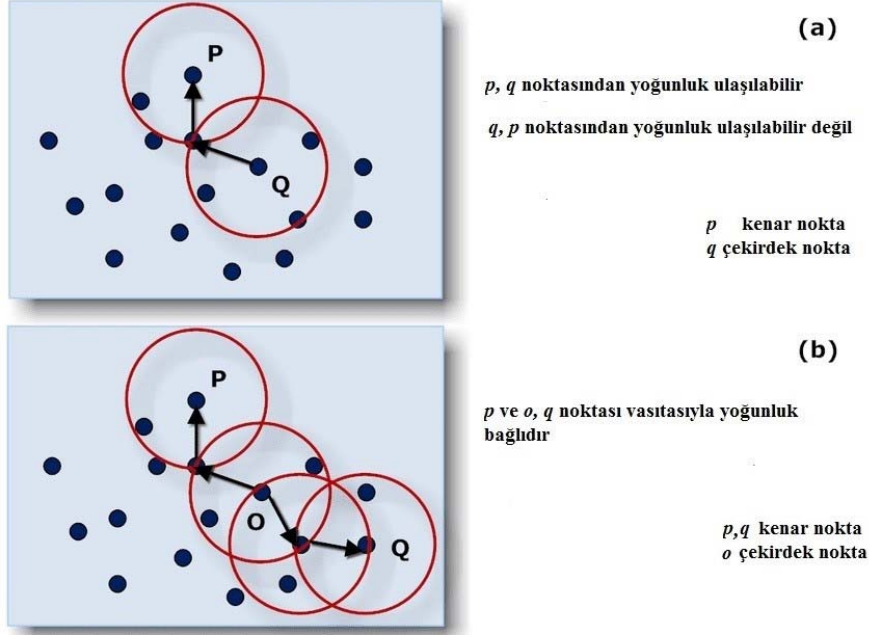
kümelere içerisindeki verilerin benzerliklerine göre ayrılmasına yardımcı olur. Çalışma kapsamında koordinat sistemi üzerinde konuma ve zamana göre binlerce veri toplanmıştır. Bu noktalar kendi aralarındaki benzerliklere ya da yakınlık derecelerine göre kümelere ayrılabilir ve kümelerin yoğunluklarına göre bir tahmin yapılabilir. Bu aşamada yoğunluk bazlı çalışan algoritmalar kullanılabilir.

Çalışma kapsamında yoğunluk bazlı algoritmalar karşılaştırılmış ve incelenmiştir (Şekil 1). Bunlardan birisi de DBSCAN (A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise) algoritmasıdır. DBSCAN algoritması yoğunluk tabanlı kümeleme algoritmasıdır [38-40]. Algoritma temel olarak çekirdek nesne, Eps, MinPts, doğrudan yoğunluk erişilebilir nokta, yoğunluğa bağlı nokta parametrelerinden oluşur. Algoritma, Eps ve MinPts değerlerini giriş parametresi olarak alır. Veritabanındaki herhangi bir nesneden başlayarak tüm nesnelere kontrol eder. Eğer kontrol edilen nesne daha önce bir kümeye dahil edilmiş ise işlem yapmadan diğer nesneye geçer. Eğer nesne daha önce kümelanmemiş ise, bir bölge sorgusu yaparak nesnenin Eps komşuluğundaki komşularını bulur. Komşu sayısı MinPts'den fazla ise, bu nesne ve komşularını yeni bir küme olarak adlandırır. Daha sonra, önceden kümelanmemiş her bir komşu için yeni bölge sorgusu yaparak yeni komşular bulur (Şekil 2). Bölge sorgusu yapılan noktaların komşu sayıları MinPts'den fazla ise kümeye dahil eder [1].

Şekil 2 üzerinde yoğunluk bazlı algoritmalarda sıkça rastlanan iki terime değinilmiştir. Bunlardan birincisi yoğunluk ulaşılabilir durumudur. Şekil 2a üzerinde görüleceği üzere  $p$  ve  $q$  noktalarını içeren bir noktalar topluluğunda yoğunluk ulaşılabilir durumlar incelenmiştir.  $Q$  noktası DBSCAN algoritmasında rastgele seçilen,



Şekil 1. Yoğunluk bazlı algoritmaların karşılaştırılması (Comparison of density-based algorithms)



**Şekil 2.** DBSCAN algoritmasının nasıl çalıştığına ilişkin görünüm (Representation of how DBSCAN algorithm Works) [41]

herhangi bir kümeye daha önce dahil edilmemiş, epsilon komşuluğunda minpoints değeri kadar (bu durumda minpoints 5 olarak varsayılıyor) nokta içeren bir durumdur. Yani  $q$  noktası yeni oluşacak olan bir kümenin başlangıcı, bir başka deyimle çekirdek noktadır. Çekirdek nesne olan  $q$  seçildikten sonra, oluşturduğu kümedeki diğer noktalarında komşuluğuna bakılarak küme genişleyecektir. Şekil 2a üzerinde görüleceği üzere,  $q$  noktasından oluşan küme genişledikçe  $p$  noktasını da içine alan bir küme haline gelecektir. Bu durumda  $p$  noktası  $q$  noktasından yoğunluk ulaşılabilir olarak değerlendirilir. Öte yandan, Şekil 2a üzerinde yer alan  $p$  noktası rastgele seçilmiş ve daha önce hiçbir kümeye dahil edilmemiş bir durum olarak noktalar topluluğu ele alınsa, epsilon komşuluğunda minpoints parametresini sağlayamadığı için bir küme oluşturamayacaktır. Dolayısıyla  $p$  ve  $q$  noktaları, eğer DBSCAN algoritması göz önüne alındığında rastgele seçilen nokta olarak  $p$  noktası seçildiğinde aynı kümeye dahil olamamaktadır. Çünkü  $p$  noktası küme başlangıcı şartlarını sağlayamamakta diğer bir deyişle çekirdek nesne olamamaktadır. Bu durumda ise  $q$  noktası  $p$  noktasından yoğunluk ulaşılabilir değildir. Şekil 2 üzerindeki diğer bir durum ise yoğunluk bağlı durumdur. Bu durum, çekirdek nokta olamayan iki noktanın çekirdek nokta olan bir başka nokta aracılığı ile aynı kümeye dahil olmasıdır. Şekil 2b üzerinde görüleceği üzere  $o$  noktası bir çekirdek noktadır.  $p$  ve  $q$  noktaları ise  $o$  noktası aracılığı ile birbirine bağlıdır, diğer bir ifadeyle aynı kümededir.

Çalışma kapsamında verinin 3. boyutu olarak zaman parametresi de bulunmaktadır. Çalışmaların çoğu, sıradan verilerden (uzamsal olmayan ve zamansal olmayan veriler) kümeleri keşfetmeye odaklanır, bu nedenle uzamsal-zamansal verileri kümelemek için kullanmak pratik değildir

[3]. Mekansal-zamansal veriler için bilgi bulma süreci mekansal olmayan ve zamansal olmayan veriler için daha karmaşıktır [3]. Çünkü uzamsal-zamansal kümeleme algoritmaları, gerekli bilgiyi elde etmek için nesnelerin uzamsal ve zamansal komşularını dikkate almak zorundadır [3]. Uzamsal veri, enlem ve boylam bilgisi, kendi aralarındaki komşulukları kilometre, metre gibi bir uzunluk birimi ile ifade edilirken, üçüncü parametre olan zaman ile bu birimlerin komşuluğu ifade edilemez. Bu yüzden uzamsal-zamansal veriyi öbekleyebilecek bir özel algoritmaya ihtiyaç duyulur. ST-DBSCAN (Spatial-Temporal Density-Based Spatial Clustering of Applications with Noise) algoritması bu amaç için kullanılabilir. ST-DBSCAN algoritması DBSCAN algoritmasına kıyasla zaman ve koordinat bilgilerine özgü öbekleme yapabilir. Farklı yoğunluktaki kümeler bulunduğu zaman, DBSCAN bazı gürültü noktalarını tespit edememektedir. ST-DBSCAN algoritması ise zaman ve koordinat bazlı bu noktaların tespitini yapabilir. Algoritma zamansal ve uzamsal olarak farklı sınır nesnelere sahip olduğundan ayrışmaları daha kolay olur [3, 38]. DBSCAN algoritması Eps ve MinPts parametrelerini kullanırken, ST-DBSCAN algoritması 4 parametreye ihtiyaç duyar [3, 39]. Eps1 parametresi, enlem ve boylam cinsinden olan uzamsal verinin gereken uzaklığıdır. Eps2 parametresi ise uzamsal olmayan verinin uzaklık mesafesini temsil eder. MinPts ise Eps1 ve Eps2 değerlerini sağlayan, küme oluşturmak için gerekli olan en az eleman sayısıdır.  $\Delta E$  ise seçilen küme ortalaması ile yeni gelen değer arasında bulunması gereken eşik değerdir. Şekil 3 üzerinde yer alan kod göz önüne alındığında  $i$ 'den  $n$ 'e kadar noktalar teker teker gezinilmektedir. Eğer  $o_i$  olarak ifade edilen sıradaki nokta herhangi bir kümeye dahil değil ise diğer noktalara olan komşuluğu değerlendirilmektedir. Eps1, Eps2 ve MinPts değerleri ile noktanın bir küme oluşturup

**Algorithm** ST\_DBSCAN (D, Eps1, Eps2, MinPts,  $\Delta\epsilon$ )

```

// Inputs:
// D={o1, o2,.....,on}      Set of objects
// Eps1 : Maximum geographical coordinate (spatial) distance value.
// Eps2 : Maximum non-spatial distance value.
// MinPts : Minimum number of points within Eps1 and Eps2 distance.
//  $\Delta\epsilon$  : Threshold value to be included in a cluster.

// Output:
// C={C1, C2,.....,Ck}      Set of clusters

Cluster_Label = 0
For i=1 to n                                //(i)
  If oi is not in a cluster Then              //(ii)
    X=Retrieve_Neighbors (oi, Eps1, Eps2)      //(iii)
    If |X| < MinPts Then
      Mark oi as noise                          //(iv)
    Else                                       //construct a new cluster (v)
      Cluster_Label = Cluster_Label + 1
      For j=1 to |X|
        Mark all objects in X with current Cluster_Label
      End For
      Push(all objects in X)                      //(vi)
      While not IsEmpty()
        CurrentObj = Pop()
        Y=Retrieve_Neighbors (CurrentObj, Eps1, Eps2)
        If |Y| >= MinPts Then
          ForAll objects o in Y                //(vii)
            If (o is not marked as noise or it is not in a cluster) and
              |Cluster_Avg() - o.Value| <=  $\Delta\epsilon$  Then
                Mark o with current Cluster_Label
                Push (o)
            End If
          End For
        End If
      End While
    End If
  End For
End Algorithm

```

**Şekil 3.** ST-DBSCAN algoritmasına ilişkin kodlar (Codes for ST\_DBSCAN algorithm) [3]

oluşturmadığına bakılmakta, küme oluşturmuyor ise gürültü olarak işaretlenmektedir. Noktaya ait küme elemanları  $X$  değişkeni içerisinde tutulmaktadır.  $X$  değişkeni içerisindeki noktaların kendilerine olan komşuluklarını bulmak adına geçici olarak başka bir değişkene *Push(all objects in X)* yöntemi ile atılmaktadır. Daha sonra, bu geçici değişken boş olana dek yani koddaki *IsEmpty()* kontrolü ile kümedeki noktalar teker teker kendi komşuluklarına bakılmaktadır. Kontroller yapılırken geçici değişkende tutulan listeden nokta elemanları *Pop()* yöntemi ile listeden alınıp, *CurrentObj* değişkeni ile işleme tabi tutulmaktadır. Algoritmaya ilişkin detaylı bilgi aşağıdaki gibidir (Şekil 3).

ST-DBSCAN algoritması ile DBSCAN algoritması arasındaki fark bir örnek üzerinden açıklanabilir. Uzamsal-Zamansal veriler enlem, boylam ve zaman verisinden

oluşmaktadır. Görüldüğü üzere uzamsal-zamansal veri 3 boyutlu bir koordinat sisteminde ifade edilseydi eğer koordinatlardan birisi enlem, diğeri boylam ve üçüncüsü ise zaman olarak yer alacaktır. Enlem ve boylam bilgileri benzer birimde iken zaman bilgileri koordinat sisteminde farklı bir birimde ifade edilmektedir. Bu kapsamda, oluşturduğumuz bu 3 boyutlu koordinat sisteminde iki nokta arasındaki uzaklığı hesaplamak isteyecek olursak, iki noktanın her bir koordinat düzelemindeki farklarının karesinin toplamının kökünü alarak bu sonuca erişilebilir. Ancak burada bir toplama işlemi yapılmakta ve toplama işlemi sadece benzer cinstekileri toplayabilir. Eğer burada oluşturulan enlem, boylam ve zamandan oluşan 3 boyutlu koordinat sistemine DBSCAN algoritması uygulanır ise benzer cinsten olmayan rakamların toplanması gibi bir durum ortaya çıkacaktır. ST-DBSCAN algoritması ise bu duruma farklı bir açıdan



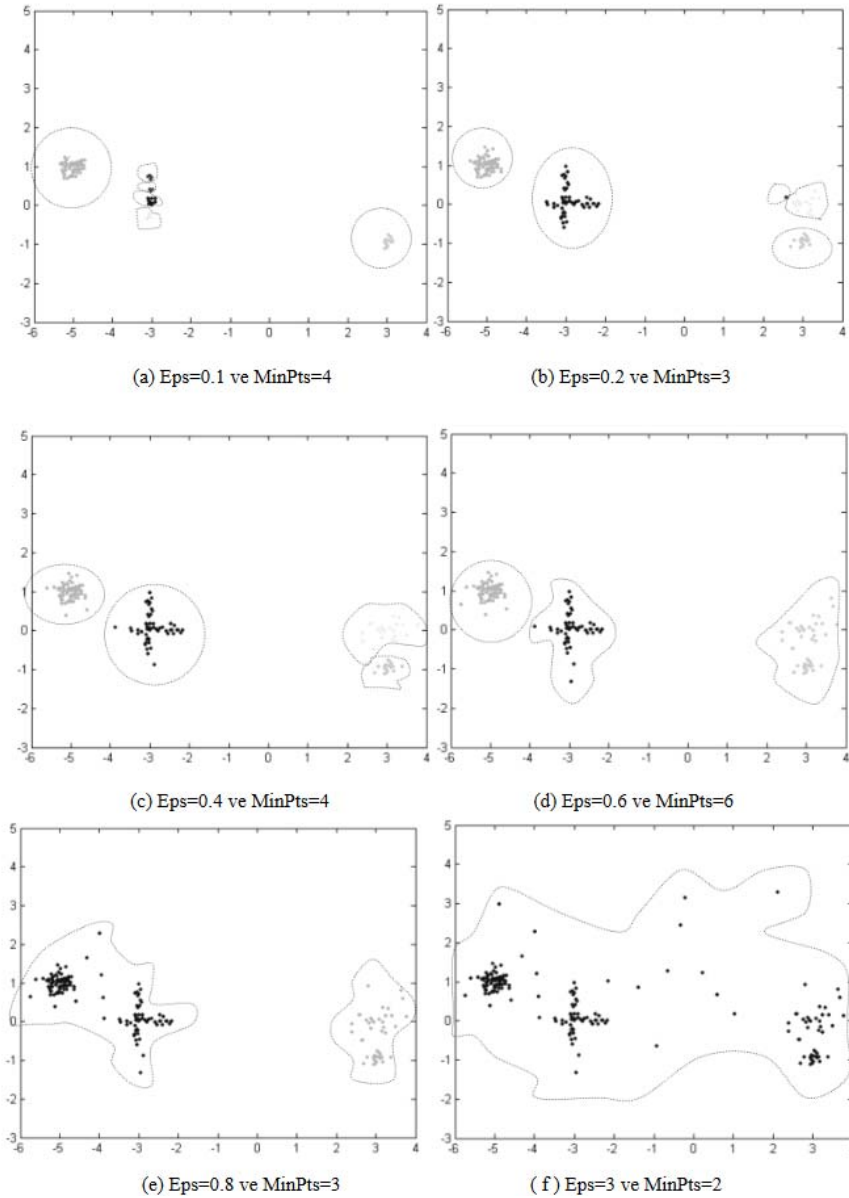
yaklaşmaktadır. Enlem ve boylam bilgileri benzer cinsteki oldukları için komşuluklarını belirleyecek olan uzaklık sadece bu değerlere uygulanmaktadır. Öte yandan, zaman verisinin uzaklığı ise kendi içerisinde hesaplanmakta ve komşuluk ilişkisi kendi özgü ayrı bir parametre ile belirlenmektedir. Bu kapsamda iki noktanın belirlediğimiz bu 3 boyutlu düzlemdeki komşulukları ise enlem, boylam ve zaman biriminin arasındaki komşuluk durumunun sağlanması ile elde edilir. Eğer enlem, boylam ya da zaman birimi komşuluklarından birisi sağlanmaz ise bu iki nokta arasındaki uzaklık komşuluğun dışındadır olarak değerlendirilmektedir.

ST-DBSCAN algoritmasındaki Eps1, Eps2 ve MinPts veya DBSCAN algoritmasındaki Eps ve MinPts girdi parametrelerinin aldıkları değerlere göre çok farklı kümeler

ortaya çıkabilmektedir (Şekil 4). Çalışma kapsamında, geçmişte toplanan veriler kullanılarak, gelecekte oluşacak veriler için bir Eps1, Eps2 ve MinPts parametrelerini tahmin edebilecek bir makine öğrenmesi metodu uygulanması amaçlanmıştır. Öncelikle geçmiş verilere ST-DBSCAN algoritması uygulanarak uygun Eps1, Eps2 ve MinPts parametreleri belirlenecek ve bu verilere ilişkin çıktılar göz önünde bulundurularak tahminler yapılacaktır.

### 3.1.1. Lineer regresyon algoritması (Linear regression algorithm)

Regresyon analizi,  $y$  olarak sembolize edilen bir bağımlı değişken ve  $x$  olarak sembolize edilen bir veya daha fazla bağımsız değişkenden oluşan nicel değerler arasındaki ilişkiyi incelemek ve tahmin etmek için kullanılan istatistiksel bir yöntemdir [42]. Regresyon analizi bir gözetimli öğrenme



Şekil 4. Farklı girdi parametrelerine göre DBSCAN algoritması çıktıları  
(DbSCAN algorithm outputs according to different input parameters)

algoritmasıdır çünkü bu yaklaşım makine öğrenimi yöntemleri ile önemli ölçüde örtüşür [1]. Basit regresyon fonksiyonunun birden fazla bağımsız değişkene ve tek bir çıktıya sahip olan geliştirilmiş yöntemine ise çoklu regresyon yöntemi adı verilir [43]. Eş. 1’de çoklu regresyon yöntemine ilişkin denklem gösterilmiştir.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k + \varepsilon \quad (1)$$

Burada  $y$  tahmin edilemeye çalışılan bağımlı değişken,  $x_1, x_2, \dots, x_k$  bağımsız değişkenlerdir.  $\beta$  ve  $\alpha$  regresyon sabiti,  $\varepsilon$  ise hata terimini ifade etmektedir. Bu çalışmada, toplanan veriler (kaç dakikada bir konum paketi gönderiyor, toplam gönderilen konum paketi sayısı) girdi olarak kullanılıp, daha sonra bunlara ilişkin çıktılarla model eğitilecek ve seçilecek konum gönderme periyoduna göre kümeleme algoritmasının parametreleri tahmin edilecektir. Cihazdan alınan girdi verilerinden olan toplam gönderilen konum paketi sayısının temel kullanım amacı, cihazın hangi yoğunlukta veri gönderdiğini tespit edebilmektir. Cihaz üzerinde bulunan GPS modülleri konum belirleyebilmek için yeterli sayıda uyduya erişemez ise konum belirleyememektedir. Elde edilen verideki, belirlenemeyen konumların çokluğu veri karakteristiğine etki edeceğinden dolayı toplam gönderilen konum paketi sayısı girdi parametresi olarak kullanılmıştır. Öte yandan cihazın kaç dakikada bir konum göndereceği bilgisi ise belirli bir konum alanındaki veri yoğunluğunu etkileyeceğinden girdi parametresi olarak seçilmiştir.

### 3.1.2. Yapay Sinir Ağları (Artificial Neural Networks)

Çalışma kapsamında değerlendirilen diğer yöntem ise yapay sinir ağlarıdır. Bunun temel sebeplerinden birisi elde edilmesi planlanan parametre tahmin probleminin doğrusal bir çözümünün olmama ihtimalidir. Yapay sinir ağlarının güçlü yanlarından birisi doğrusal olmayan problemlere çözüm getirebilmesidir [42, 44-46]. Yapay zeka yöntemlerinden birisi olan yapay sinir ağları, insan sinir hücrelerini (nöron) modelleyerek çözüm üretmekte olup geleneksel yöntemler dışında problemi kendi kendine öğrenebilmesi ve problemin sahip olduğu bilgiler arasında ilişkiler oluşturabilme yeteneğine sahiptir [42]. Temel amacı, girdi değerlerine karşılık çıktı değerlerini tahmin edebilmektir [42, 44].

İnsan sinir hücrelerinde olduğu gibi yapay sinir ağlarında da bazı yapılar vardır. Bunlar girdiler, ağırlıklar, toplama fonksiyonu, aktivasyon fonksiyonu ve çıktılarıdır. Girdiler, dışardan gelen veriyi ağlara ileten kısımdır. Ağırlıklar, nörona gelen bilginin problem için önemini gösterir ve nöron üzerindeki etkisini göstermektedir [42]. Ağırlık değerinin küçük olması girişin önemsiz olduğunu gösterir iken ağırlık değerinin büyük olması girişin önemli olduğunu gösterir [42]. Toplama fonksiyonu, gelen tüm girdilerin ağırlıkları ile çarpıp toplayarak net girdisini hesaplayan kısımdır. Aktivasyon fonksiyonu, toplama fonksiyonu tarafından iletilen net bilgiyi işleyip, çıktıya dönüştüren fonksiyondur. Problemin doğrusal olmayan çözümü için genellikle sigmoid gibi türevi alınabilen doğrusal olmayan bir fonksiyon seçilir.

Çıktı ise aktivasyon fonksiyonu ile elde edilen sonucu veren birimdir [42]. Yapay sinir ağlarının işleyişi, girdilerden alınan verinin, toplama fonksiyonu aracılığı ile net değerinin üretilmesi ile başlar. Üretilen net değer, aktivasyon fonksiyonu tarafından çıkış değerine dönüştürülür. Girişe denk gelen çıkış değeri ile elde edilen değer karşılaştırılarak hata bulunur. Her seferinde bu döngü tekrarlanır, tek fark ise ağırlıkların güncellenmesidir. Bu tekrarlama işlemi iterasyon ya da dönem olarak isimlendirilir. Böylece elde edilen hata en aza indirilmeye çalışılır [44]. Ağırlıkların değişim miktarını ifade eden öğrenme katsayısı ya da öğrenme hızıdır.

Yapay sinir ağ modelleri nöronları içeren katmanlardan oluşmaktadır. Katmanlar girdi katmanı, çıktı katmanı ve gizli katmanlardan olabilir [47]. Sadece girdi ve çıktı katmanı olan yapay sinir ağ modellerine tek katmanlı sinir ağları diğer bir adıyla perceptron olarak isimlendirilir. Girişler ve onlara karşılık gelen ağırlık değerlerinin çarpımları toplamından oluşan net değeri, eşik değeri aştığı zaman çıkış üretilir [40]. Tek ağırlık katmanı olan algılayıcı girdinin yalnızca doğrusal işlevlerini öğrenebilir ve doğrusal olmayan durumu gerçekleyemez [4]. Burada her saklı birim bir algılayıcıdır. İleri beslemeli ağlar katmanlardaki nöronlar aynı katmandaki nöronlara bağlı değil iken, bir sonraki katmandaki tüm nöronlara bağlıdır. Çok katmanlı perceptron (MLP) ileri beslemeli ağlara örnektir. Çok katmanlı perceptronlar her dönem sonucunda elde edilen hatanın, belli algoritmalar ile ağırlıklara yansıtılması ile öğrenme işlemini gerçekleştirir ve buna hatayı geriye yayma (backpropagation) denir [47]. Sinir ağlarında gözetimli öğrenme işlemi için girdiler ve bunlara denk gelen çıktı bilgileri sağlanmalıdır. Eğitim sırasında kullanılan bu veriler eğitim seti ya da eğitim kümesi olarak adlandırılır. Elde edilen öğrenmenin başarımını ölçmek için eğitim kümesinden bağımsız ve ağırlık karşılaştımadığı değerler verilerek ağ test edilir. Bu veri kümesi test kümesi ya da test seti olarak adlandırılır [42]. Bunlara ek olarak, ağırlık eğitimi sırasında genellemenin kötüleştiğini yani aşırı eğitimi tespit edebilmek adına doğrulama ya da sınav kümesi adı verilen bir veri seti eğitim seti içerisinde belli bir oranda seçilebilir. Doğrulama kümesi ile ağ eğitimi sırasında öğrenme hatası ve doğrulama hatası karşılaştırılabilir. Bu sayede belli bir dönemden sonra, ağırlık eğitildiği tespit edilip, bu dönem noktasına ağ sonlandırılabilir. Bu aynı zamanda erken durma olarak bilinir. Bu aşamadan sonra elde edilen konum ve zaman kümeleri ister öngörü için kullanılabilir iken istenilir ise anlık gelen verilerin bu desenlere uyup uymaması durumunda anormallik tespiti olarak kullanılabilir. Bu aşamada anormallik tespitine ilişkin aşağıdaki hususlara dikkat edilmelidir.

- Her normal davranışı yakalayan bir normal bölge tanımlamak zordur ve normal bölge ile anormallik bölge arasında keskin bir sınır yoktur [6].
- Anormallik kavramı uygulama alanına göre değişir. Bir alanda geliştirilen tekniği diğer bir alana taşımak çok kolay değildir [6].

- Çoğu zaman model kurmak için etiketlenmiş eğitim ve doğrulama verilerini bulmak zordur [6].
- Anormallikler normalmiş gibi görülebilir. Bu durum normal davranışı tanımlamayı daha da zorlaştırır [6].
- Çoğu çalışma alanında normal davranış evrildiği için başta yapılan normal tanımın temsili yeterli olmayabilir [6].

#### 4. SONUÇLAR VE TARTIŞMALAR (RESULTS AND DISCUSSIONS)

Çalışma kapsamında toplam 4 haftalık süre içerisinde 5558 adet konum veri paketi toplanmıştır. Cihaz verisi içerisinde cihazın kendi hatasından kaynaklı hatalı veri paketleri çalışmaya dahil edilmemiştir (Tablo 2).

Veri hazırlık aşamasında verilerin 5 dakika aralıklarla toplandığı göz önüne alındığında ortalama veri toplama süresinin 1 aydan fazla sürdüğü görülmüştür. Çalışma kapsamında düşünülen 5, 10, 15, 20, 25, 30 ve 60 dakika aralıkları düşünüldüğünde ise 7 aylık bir sürece denk gelmektedir. Veri toplama süreci, 5 dakika aralıklarla elde edilen konum veri paketlerinin azaltılması ile kısaltılmıştır. Azaltılmış verilerden diğer zaman aralıkları elde edilmiştir (Tablo 3).

Tablo 3'te görüleceği üzere veri gönderme sıklığı 5 dakika olan verilerden 10 dakikalık veriler elde edildiğinde elde edilen rakam tam olarak 5 dakikalık verinin yarısı değildir. Bunun temel sebebi veri seyreltme işleminden kaynaklanmaktadır. Cihaz üzerindeki GPS modül konum

belirlemediğinde gönderilecek paketin süresi de uzamaktadır. Şekilde görüleceği üzere bir cihazın 60 dakikalık zaman ekseninde 5 dakika veri gönderme sıklığına ayarlandığında toplam 7 paket (15, 23, 31, 37, 42, 49, 56) gönderdiği örnek olarak verilmiştir (Şekil 5). Cihaz konumları belirlemeye çalışıldığında, 15. dakikaya kadar konum belirleyememiştir. Bu veriler seyreltilirken 15. dakika verisi 10 dakikalık verinin ilki olarak alınmaktadır. Daha sonra üzerine 10 dakika eklenip (yani 25. dakikadan sonra), ilk sıradaki veri 10 dakikalık verinin ikinci paketi (31. dakikadaki veri paketi) olarak alınmıştır. İşlemler bu şekilde devam ettiğinde 5 dakika veri gönderme sıklığı toplam paket sayısı 7 iken 10 dakika veri gönderme sıklığı toplam paket sayısı 4 olmaktadır. Bu yüzden tabloda gösterilen sayılar veri gönderme sıklığı 2 katına çıksa dahi toplam paket sayısı ters orantılı olarak tam olarak yarısına düşmemektedir (Tablo 3).

**Tablo 3.** Zaman aralıklarına göre konum veri paketi sayısı  
(Number of location data packets by time intervals)

Hafta Sayısı	Veri Gönderme Sıklığı	Veri Paketi Sayısı
4	5	5558
4	10	2894
4	15	1969
4	20	1693
4	25	1407
4	30	1193
4	60	635

Toplanan 4 haftalık veriden 2 haftalık verilerin çıkarılması amacıyla 1 haftalık bölümlere ayrılan konum paketlerinden

**Tablo 2.** Takip cihazından elde edilen konum veri paketi örnekleri  
(Location data packet samples obtained from the tracking device)

Cihaz Adı	IMEI	Olay Kodu	Pil (%)	Zaman	Enlem	Boylam	Hata	Kaynak	Adres Bilgisi
T01-0001	-	Bağlantı Yenilendi	80	21.06.2019 09:25:47	39.87572	32.86401	0	GPS	508. Cd. Sancak Merkez Ankara / Türkiye Katar Cd.
T01-0001	-	Bağlantı Yenilendi	75	21.06.2019 09:19:29	39.87763	32.86763	0	GPS	Birlik Merkez Ankara / Türkiye 508. Cd.
T01-0001	-	Bağlantı Yenilendi	80	21.06.2019 09:13:16	39.87574	32.86383	0	GPS	Sancak Merkez Ankara / Türkiye Turan Güneş Blv. Sancak
T01-0001	-	Bağlantı Yenilendi	75	21.06.2019 09:07:02	39.87578	32.86368	0	GPS	Merkez Ankara / Türkiye 424. Sk.
T01-0001	-	Bağlantı Yenilendi	75	21.06.2019 09:00:38	39.87834	32.86989	0	GPS	Birlik Merkez Ankara / Türkiye

2 haftalık farklı veri kümeleri elde edilmiştir. Her veri toplama sıklığı için 2 haftalık toplamda 6 farklı veri kümesi elde edilmiştir. Örnek olarak birinci hafta verileri ile dördüncü hafta verileri birleştirilerek bir veri kümesi oluşturulmuştur. Bu kombinasyon tüm veri gönderme sıkları için gerçekleştirilmiştir (Tablo 4).

**Tablo 4.** Beş dakika aralıkla toplanan veriye ilişkin 2 haftalık veri paketi sayısı

(Number of 2-week data packets for data collected at a five-minute interval)

Haftalar	Veri Gönderme Sıklığı	Veri Paketi Sayısı
1. hafta – 2.hafta	5	2760
2. hafta – 3. hafta	5	2747
2. hafta- 4. hafta	5	2839
3. hafta- 4. hafta	5	2798
1. hafta- 4. hafta	5	2811
1. hafta- 3. hafta	5	2719

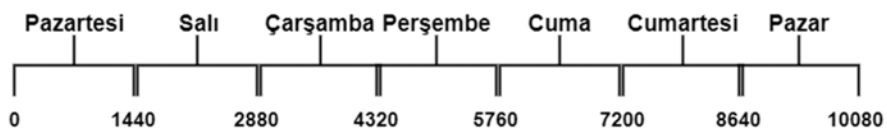
Bu çalışmada takip cihazından alınan verilerin diğer takip cihazlarında da ortak olarak elde edilebilecek konum ve zaman verisine odaklanılmıştır. Kullanılan algoritmalarda bu verilerden, konum verisine ilişkin enlem ve boylam bilgisi değiştirilmeden kullanılabilirken, zaman verisinin kullanılabilmesi için düzenlenmesi gerekmektedir. Bu ihtiyaç, çalışma sonucunda elde edilecek haftanın günlerine ilişkin zaman aralıklarının kesintisiz şekilde komşu olması ile ilgilidir. Bu yüzden veri ilk önce haftanın günlerine göre düzenlenmiştir. Veri analizinde, 1 günün 24 saat ile sonlanması ile yeniden başa dönmelerinden dolayı problemler ortaya çıkmıştır. Bu kapsamda haftanın başlangıcı Pazartesi 00:01 saati 1. dakika olarak alınmış ve her geçen dakika üzerine eklenerek bir haftada 10,080 dakika olacak şekilde

zaman verisi düzenlenmiştir. Böylece zaman komşuluklarının tespiti kolaylaşmıştır (Şekil 5). İlk olarak enlem, boylam ve haftalık dakika bilgisi içeren veri paketleri, yapılacak işlemlere hazır hale getirilmiştir. Hazır hale getirilen verilere ST-DBSCAN algoritması uygulanmış ve uygulama sırasında ST-DBSCAN algoritmasının girdileri olan Epsilon1, Epsilon2 ve MinPoints parametrelerinin seçimine dikkat edilmiştir. ST-DBSCAN algoritmasının uygulanması ile veriler üzerinden elde edilen konum kümeleri incelenmiştir. İnceleme sonucunda, algoritmanın kümeleri ve gürültüleri düzgün şekilde tespit ettiği anlaşılmıştır (Şekil 6). Şekil 7 de, 5 dakika veri gönderme sıklığı toplanmış veriler üzerine Epsilon1, Epsilon2 ve MinPoints parametrelerinin farklı değerlerde nasıl değiştiği gösterilmiştir. Şekil 7 üzerinde yer alan (a) kısmında 5 dakika veri gönderme sıklığı için en uygun parametreler verilmiş olup, takip edilen nesneye ait tüm rutinler düzgün şekilde elde edilmiştir. Şekil 7 üzerinde yer alan (b) kısmında ise Epsilon1, Epsilon2 ve MinPoints parametreleri değiştirilmiş olup, tespit edilmesi gereken tüm rutinlerde bir takım kayıplar yaşanmıştır. Öte yandan, Epsilon1, Epsilon2 ve MinPoints parametreleri Şekil 7 üzerinde yer alan (c) kısmında uygun değerlerinden uzaklaştıkça, tespit edilmesi gereken rutinlerin çoğunu kaybetmiştir.

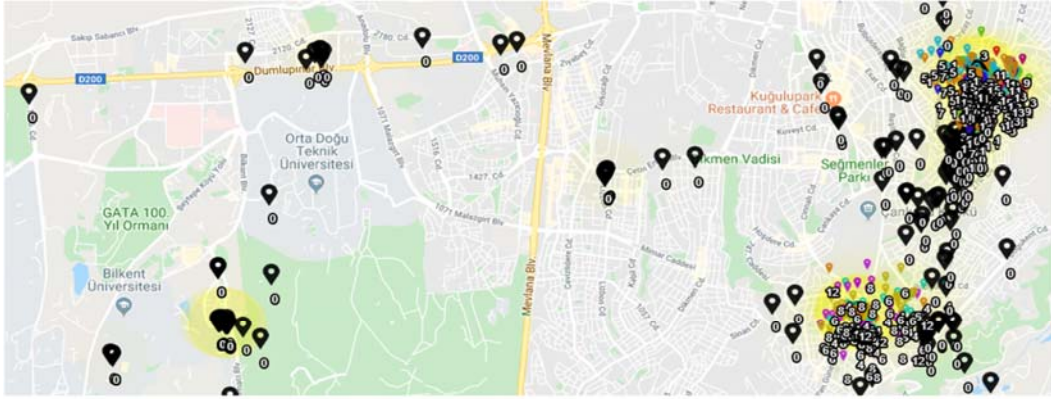
Daha sonra 2 haftalık kombinasyonlar şeklinde oluşturulan konum verileri üzerinde ST-DBSCAN algoritması uygulanmış olup, takip edilen nesnenin konum davranışlarını düzgün şekilde ortaya koyan girdi parametreleri toplanmıştır. 2 haftalık veriler üzerinden elde edilen rutinlere göz atıldığında, takip cihazı taşıyan kişiye ait evde zaman geçirme rutinleri ile işte zaman geçirme rutinleri tam olarak tespit edilebilmiştir. ST-DBSCAN algoritmasının girdi parametrelerinden birisi olan, iki nokta arasındaki zaman farkı, genel olarak takip cihazının veri gönderme sıklığının 2



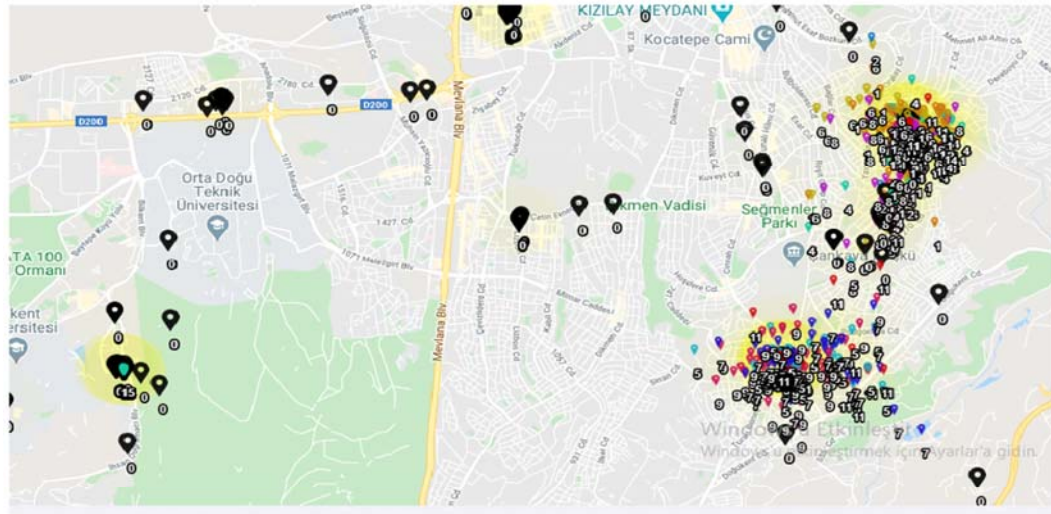
**Şekil 5.** Beş dakikalık veri kümesinden 10 dakikalık veri elde yöntemi  
(10-minute data acquisition method from a 5-minute dataset)



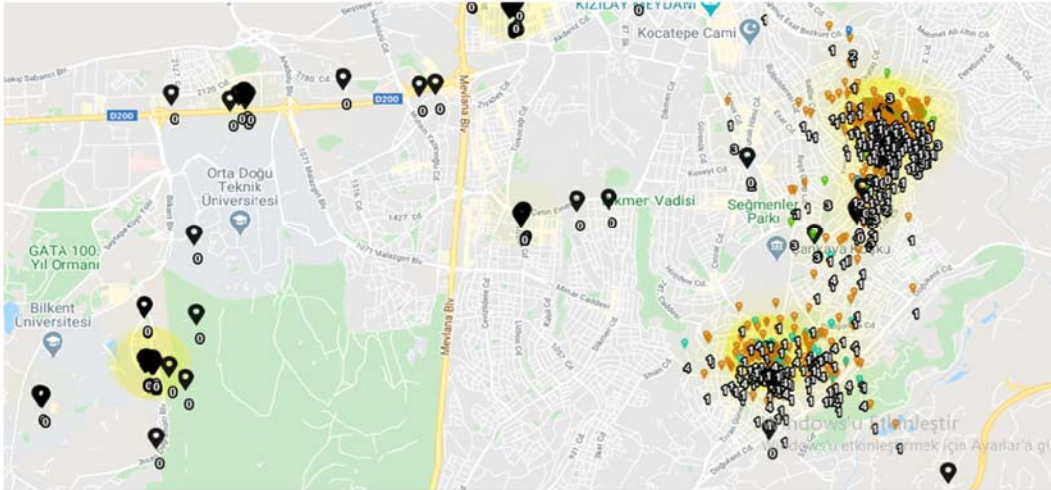
**Şekil 6.** Dakika bazlı haftalık konum verisinin gösterimi (Representation of minute-based weekly location data)



(a) Epsilon1 = 800, Epsilon2 = 20, MinPoints = 8



(b) Epsilon1 = 1500, Epsilon2 = 30, MinPoints = 10



(c) Epsilon1 = 2000, Epsilon2 = 40, MinPoints = 15

Şekil 7. ST-DBSCAN algoritmasının farklı parametrelere göre sonuçlarının harita üzerinde gösterimi  
(Map representation of the results of ST-DBSCAN algorithm with different parameters)

ile 4 katı şeklinde seyretmiştir. Tespit edilen rutin saatlerde girdi parametresi kadar fark olabileceği görülmüştür. Bu

parametrelerin belirlenmesi sırasında oluşan konum davranışı kümeleri, takip cihazı taşıyan kişi tarafından onayı

alınarak etiketlenmiştir. Toplanan 4 haftalık verinin 2 haftalık kombinasyonları şeklinde 6 farklı bilgi elde edilmiştir. Bu verilerin 5, 10, 15, 20, 25, 30 ve 60 dakika aralıklarla veriler gönderen varyasyonları da göz önüne alındığında, 42 farklı ST-DBSCAN algoritması girdisi etiketlenmiştir.

Tablo 5 üzerinde 5 dakika aralıklar ile toplanmış birinci ve dördüncü hafta verilerine ST-DBSCAN algoritması uygulanmış olup, algoritma girdileri olan Epsilon1 değeri 800 seçilmiş olup, Epsilon2 değeri 20 ve MinPoints ise 8 olarak seçilmiştir. Bu işlem sonrasında elde edilen sonuç cihazı taşıyan nesneye ait rutinleri ortaya çıkarmıştır. Tablo 5 üzerinde küme numaraları nesneye ait rutinleri numaralandırır iken, başlangıç zamanı ve bitiş zamanı bu rutinlerin hangi gün ve saat aralıklarında olduğu göstermektedir. Öte yandan, Tablo 5 üzerindeki enlem, boylam ve açık adres bilgisi kısmı ise belirlenen rutinlerin alansal olarak merkez noktasını göstermektedir. Verilerin etiketlenmesi sırasında, doğru bilgilerinin elde edilip edilmediğinin kontrol edilmesi sırasında kolaylık olması açısından koordinat bilgileri ve açık adres bilgileri verilmiştir. Örnek olarak 13 numaralı rutin (diğer bir deyişle küme) pazartesi günü 07:59 ile 17:44 arasında ve açık adres olarak "508. Cd. Sancak Merkez Ankara / Türkiye" konumundadır. Veriler etiketlenir iken takip cihazını taşıyan

nesnenin bu dönemde bu aralıkta olup olmadığı bu şekilde kontrol edilmiştir.

Takip edilen nesneye ait rutinleri, Tablo 5 üzerinde görüldüğü üzere ST-DBSCAN algoritması başarılı şekilde tespit edebilmiştir. Tablo 5 üzerindeki veri elde edilirken girdi olarak Epsilon1 parametresi 800, Epsilon2 parametresi 20 ve MinPoints parametresi 8 olarak belirlenmiştir. Çıktı olarak nesnenin konum ve zaman açısından rutinleri elde edilmiştir. Veri gönderme sıklığı (5, 10, 15, 20, 25, 30, 60 dakika) ve hafta kombinasyonlarından oluşan 42 farklı veri kümeleri teker teker Epsilon1, Epsilon2 ve MinPoints girdi parametreleri ile etiketlenmiştir.

ST-DBSCAN algoritmasının girdi parametreleri olan veri gönderme sıklığı ve veri paketi sayısına göre çıktı parametreleri üzerinde bir ilişki oluşmuştur (Tablo 6). Girdi parametreleri ile çıktı parametreleri arasındaki ilişki, her bir çıktı parametresi için farklıdır. Bu yüzden Epsilon1, Epsilon2 ve MinPoints parametreleri için farklı modeller geliştirme ihtiyacı ortaya çıkmıştır. Veri gönderme sıklığı ve veri paketi sayısı girdi parametrelerine normalleştirme yöntemi uygulanmıştır. Normalleştirme yöntemi kapsamında, ortalama değer (mean value) ve standart sapması belirlenen girdi parametrelerinin standart skoru bulunmuş ve diğer adımlar bu değerler üzerinden

**Tablo 5.** Beş dakika aralıklarla toplanmış 1. hafta ve 4. hafta verileri üzerinde ST-DBSCAN algoritmasının tespit ettiği rutinler (The routines detected by ST-DBSCAN algorithm on the first and fourth week data collected at a five-minute interval)

Küme Numarası	Enlem	Boylam	Başlangıç Zamanı	Bitiş Zamanı	Açık Adres Bilgisi
1	39.90777	32.88257	Pazar 21:39	Pazartesi 07:56	244. Cd. Aşıkpaşa Merkez Ankara / Türkiye
2	39.90773	32.88234	Pazar 00:02	Pazar 15:12	Vedat Dalokay Cd. Aşıkpaşa Merkez Ankara / Türkiye
3	39.90762	32.88261	Cumartesi 21:19	Cumartesi 23:59	244. Cd. Aşıkpaşa Merkez Ankara / Türkiye
4	39.90603372	32.88077957	Cuma 18:58	Cumartesi 08:22	Vedat Dalokay Cd. Aşıkpaşa Merkez Ankara / Türkiye
5	39.87973	32.86152	Cuma 08:00	Cuma 17:57	Simon Bolivar Blv. Çankaya Merkez Ankara / Türkiye
6	39.90469559	32.88434387	Perşembe 18:01	Cuma 07:54	Şemsettin Günaltay Cd. Aşıkpaşa Merkez Ankara / Türkiye
7	39.8757	32.86401	Perşembe 08:13	Perşembe 17:48	508. Cd. Sancak Merkez Ankara / Türkiye
8	39.90769	32.88245	Çarşamba 19:51	Perşembe 08:12	Vedat Dalokay Cd. Aşıkpaşa Merkez Ankara / Türkiye
9	39.87921	32.8603	Çarşamba 08:06	Çarşamba 17:52	727. Sk. Yıldızevler Merkez Ankara / Türkiye
10	39.90775644	32.88251792	Salı 18:03	Çarşamba 07:53	Vedat Dalokay Cd. Aşıkpaşa Merkez Ankara / Türkiye
11	39.87550986	32.86393886	Salı 08:09	Salı 17:50	507. Sk. Sancak Merkez Ankara / Türkiye
12	39.90755	32.88247	Pazartesi 18:02	Salı 07:58	Vedat Dalokay Cd. Aşıkpaşa Merkez Ankara / Türkiye
13	39.87578878	32.86389119	Pazartesi 07:59	Pazartesi 17:44	508. Cd. Sancak Merkez Ankara / Türkiye



gerçekleştirilmiştir. Her bir çıktı parametresi için lineer regresyon ve ileri beslemeli yapay sinir ağlarından olan çok katmanlı perceptron (MLP) karşılaştırılarak incelenmiş ve en uygun model seçimi için karşılaştırılmıştır. Çok katmanlı perceptron olarak 1 ve 2 gizli katmana sahip ve her bir katmanında 50 sinir hücresi bulunan bir sinir ağı seçilmiştir. Hem lineer regresyon hem de çok katmanlı perceptron kullanılırken tahmin performans ölçütü olarak ortalama karesel hata (MSE) seçilmiştir. Çok katmanlı perceptron ağı içerisinde aktivasyon fonksiyonu olarak sigmoid fonksiyonu seçilmiştir. Çıktı parametrelerine ilişkin modeller oluşturulurken, uygun dönem (epoch) ve öğrenme hızı (learning rate) sayılarının bulunabilmesi için öncelikle, epoch sayısı 2000 olarak belirlenmiştir. Test veri kümesi olarak 1.hafta – 4.hafta veri kombinasyonu ile 1. hafta – 3. hafta veri kombinasyonu seçilmiştir. Toplamda 14 farklı etiketlenmiş veriyi içermektedir. Eğitim kümesi olarak diğer ikili hafta kombinasyonları seçilmiştir. Eğitim kümesinin büyüklüğü ise 28 farklı etiketlenmiş veriden oluşmaktadır. Öğrenme işlemi sırasında eğitim kümesi 80-20 oranında olacak şekilde öğrenme ve test verisi olarak kullanılmıştır.

Epsilon1 çıktı parametresi için farklı öğrenme hızlarında, epoch sayısı 2000 olacak şekilde lineer regresyon ve çok katmanlı perceptron modelleri eğitime tabi tutulmuştur. Her bir eğitim işlemi çıktıları ile birlikte listelenmiştir (Tablo 7).

Eğitim çıktılarının ortalama karesel hata grafikleri incelenmiştir. Elde edilen ortalama karesel hata doğrulama

değerinin mümkün olduğunca ortalama karesel hata öğrenme değerine yakın olmasına dikkat edilmiştir. Uygun değerler seçilirken ortalama karesel hata doğrulama değerinin, ortalama karesel hata öğrenme değerini geçtiği nokta olan epoch değerleri alınmıştır. Ortalama karesel hata doğrulama rakamının diğer rakamı geçmediği ve belli bir epoch sayısından sonra rakamların birbirine yaklaşmalarının yavaşladığı noktalar uygun epoch sayısı olarak kabul edilmiştir. Modeli etkileyecek kadar bir değişim gözlenmediği durumlarda eğitim işlemi bitirilmiştir.

Epsilon1 parametresi için en uygun değerlere bakıldığında 1 gizli katmana sahip çok katmanlı perceptron modelinin hem ortalama karesel hata öğrenme – ortalama karesel hata doğrulama arasındaki fark hem de ortalama karesel hata test sayısı bakımından diğer modellerden daha iyi sonuçlar verdiği gözlenmiştir (Tablo 8).

Epsilon1 parametresi için elde edilen öğrenme modelleri 3 boyutlu düzlem üzerinde gösterilmiştir. Şekil 7 de görüleceği üzere en solda yer alan düzlem bilgisi lineer regresyon modeline aittir. Şekilde yer alan mavi noktalar ise eğitim verisinde kullanılan etiketlenmiş verilerdir. Lineer regresyon, düzleminin çoğu noktasına temas dahi edemediği ve elde edilen ortalama karesel hata test rakamının da göz önüne alındığında doğrusal olmayan bir çözüm gerektirdiği gözükülmektedir. En sağda yer alan şekil ise 2 gizli katmanlı çok katmanlı perceptron modeline aittir. Elde edilen düzlem doğrusal olmayan bir çözüm olsa da ortalama karesel hata

**Tablo 6.** Birinci hafta - dördüncü hafta verileri üzerinde etiketlenen ST-DBSCAN girdi parametreleri (ST-DBSCAN input parameters tagged on first and fourth week data)

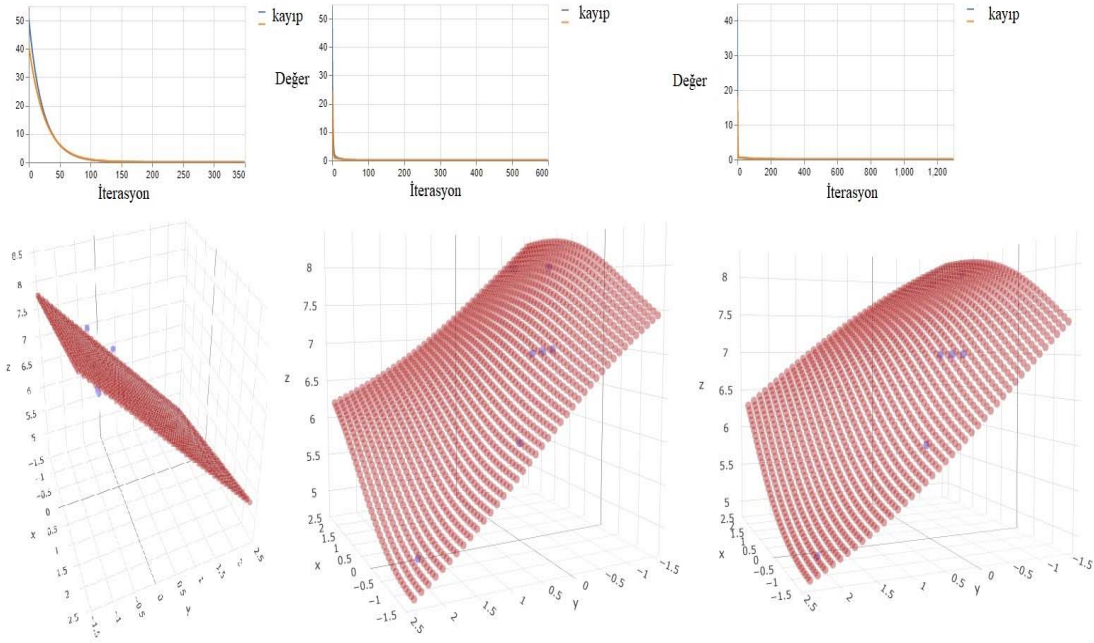
Veri Gönderme Sıklığı	Veri Paketi Sayısı	Epsilon1	Epsilon2	MinPoints
5	2811	800	20	8
10	1458	800	35	8
15	988	1500	50	7
20	830	2000	60	7
25	700	1000	65	7
30	605	800	75	6
60	317	800	140	5

**Tablo 7.** Epsilon 1 parametresi için model sonuçları (Model results for the Epsilon 1 parameter)

Model	Dönem	Öğrenme Hızı	Ortalama Karesel Hata Öğrenme	Ortalama Karesel Hata Test	Ortalama Karesel Hata Doğrulama
Lineer Regresyon	2000	0.01	120689.718	143814.34	87519.234
Lineer Regresyon	2000	0.001	125153.710	160592.890	91029.031
Lineer Regresyon	2000	0.0001	625451.875	764018.687	586725.625
1 Gizli Katman	2000	0.01	111643.992	132687.281	84659.296
1 Gizli Katman	2000	0.001	22599.337	18572.791	24141.367
1 Gizli Katman	2000	0.0001	49646.355	52899.167	19508.382
2 Gizli Katman	2000	0.01	137500.031	180238.109	111388.906
2 Gizli Katman	2000	0.001	124365.593	160012.359	101006.593
2 Gizli Katman	2000	0.0001	124425.734	160280.453	101052.671

**Tablo 8.** Epsilon1 parametresine ilişkin uygun değerler (Appropriate values for the epsilon1 parameter)

Model	Dönem	Öğrenme Hızı	Ortalama Karesel Hata Öğrenme	Ortalama Karesel Hata Test	Ortalama Karesel Hata Doğrulama
Lineer Regresyon	500	0.01	120852.9375	146024.953125	87930.3671875
1 Gizli Katman	1600	0.001	23497.578125	19417.998046875	23674.9140625
2 Gizli Katman	800	0.0001	128729.8984375	171832.953125	101036.1875

**Şekil 8.** Epsilon1 parametresi için ortalama karesel hata grafiği (Average square error graph for parameter Epsilon1)

öğrenme – ortalama karesel hata doğrulama arasındaki farkın istenilen epoch sayısında kapanmadığı ve öğrenme işleminin eksik tamamlandığı gözlemlenmiştir. Şekil 8 de ortada yer alan kısım 1 gizli katmanlı çok katmanlı perceptron modeline aittir. Elde edilen düzlem büyük oranda öğrenme verilerine yakın ve aynı zamanda ortalama karesel hata öğrenme – ortalama karesel hata doğrulama arasındaki fark diğer modellere göre çok daha düşüktür. Bu şartlar altında Epsilon1 parametresinin tahmini için 1 gizli katmana sahip çoklu katman perceptron modelinin daha uygun olduğu sonucuna varılmıştır.

Tablo 9 üzerinde kullanılan veriler başlığı altında 2 hafta boyunca toplanan konum verilerinin hangi haftalara ait olduğu yer almaktadır. Veri gönderme sıklığı ve veri paketi sayısı çok katmanlı algıyıcılar için kullanılan özelliklerdir (features). Etiketli veriler Epsilon1 başlığı altında ise daha önceden ST-DBSCAN algoritması için etiketlenmiş veri (labeled data) yer almaktadır. Doğrusal regresyon, 1 gizli katman sahip MLP ve 2 gizli katmana sahip MLP başlıkları altında ise yapay sinir ağlarına girdi verileri verildiğinde çıktı olarak elde edilen değerler yer almaktadır. Epsilon2 çıktı parametresi de farklı öğrenme hızlarında epoch sayısı 2000 olacak şekilde eğitime tabi tutulmuştur. Her bir işlem

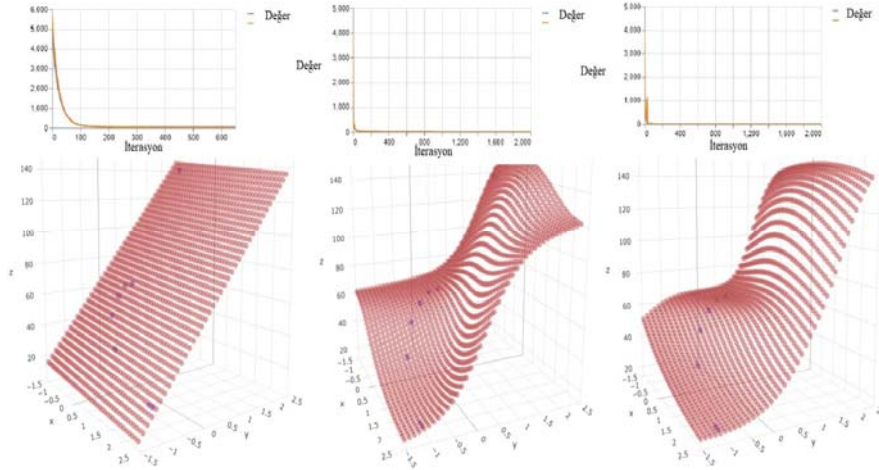
için eğitim çıktıları listelenmiştir (Tablo 10). Eğitim işlemleri sonrasında ortalama karesel hata grafikleri incelenmiştir. Ortalama karesel hata grafiği üzerinde eğitim rakamlarının yavaşladığı ya da ortalama karesel hata öğrenme değeri ile ortalama karesel hata doğrulama değerlerinin birbirine yaklaştığı noktadaki epoch değerleri uygun kabul edilmiştir.

Epsilon2 çıktı parametresi için lineer regresyon model performansının yapay sinir ağları modellerine göre daha kötü olduğu görülmüştür (Tablo 11). 1 gizli katmana sahip çok katmanlı perceptron modelinde ortalama karesel hata test değerinin en düşük değerde olmasına karşın ortalama karesel hata öğrenme ile ortalama karesel hata doğrulama değerleri, 2 katmana sahip çok katmanlı perceptron modelinden daha yüksek değerlere sahip olduğu gözlemlenmiştir. 2 katmana sahip çok katmanlı perceptron modelinin en düşük doğrulama değerine sahip olduğu görülmüştür (Tablo 10).

Epsilon 2 değeri için en solda lineer regresyon modeli, ortada 1 gizli katmana sahip çok katmanlı perceptron modeli ve en sağda ise 2 gizli katman sahip çok katmanlı perceptron modeline ait ortalama karesel hata grafiği ve 3 boyutlu ortamda elde edilen düzlem kesitleri Şekil 9 üzerinde

**Tablo 9.** Etiketli Epsilon1 parametresi ile modellerden elde edilen sonuçların karşılaştırması  
(Comparison of the results obtained from the models with the labeled Epsilon1 parameter)

Kullanılan Veriler	Veri Gönderme Sıklığı	Veri Paketi Sayısı	Etiketli Veriler Epsilon1	Doğrusal Regresyon	1 Gizli Katmana Sahip MLP	2 Gizli Katmana Sahip MLP
1. hafta ve 2. hafta	5	2760	800	772	844	1145
1. hafta ve 2. hafta	10	1446	800	1111	844	1145
2. hafta ve 3. hafta	5	2747	800	776	844	1145
2. hafta ve 3. hafta	10	1436	800	1114	844	1145
2. hafta ve 4. hafta	15	1000	800	1184	1212	1145
2. hafta ve 4. hafta	20	837	1500	1170	1714	1145
3. hafta ve 4. hafta	15	989	1500	1187	1263	1145
3. hafta ve 4. hafta	20	828	1500	1173	1711	1145
3. hafta ve 4. hafta	25	692	800	1151	916	874

**Şekil 9.** Epsilon2 parametresi için ortalama karesel hata grafiği (Average square error graph for parameter Epsilon2)

gösterilmiştir. Lineer regresyon modeli hariç diğer modeller birbirine çok yakın sonuçlar verdiği görülmektedir. Bu kapsamda, Epsilon2 parametresi için 2 katmana sahip çok katmanlı perceptron modelinin en uygun seçenek olduğu sonucuna verilmiştir. Tablo 12 üzerinde kullanılan veriler başlığı altında 2 hafta boyunca toplanan konum verilerinin hangi haftalara ait olduğu yer almaktadır. Veri gönderme sıklığı ve veri paketi sayısı çok katmanlı algıyıcılar için kullanılan özelliklerdir (features). Etiketli veriler Epsilon2 başlığı altında ise daha önceden ST-DBSCAN algoritması için etiketlenmiş veri (labeled data) yer almaktadır. Doğrusal regresyon, 1 gizli katman sahip MLP ve 2 gizli katmana sahip MLP başlıkları altında ise yapay sinir ağlarına girdi verileri verildiğinde çıktı olarak elde edilen değerler yer almaktadır. MinPoints parametresi de farklı öğrenme hızlarında ve epoch sayısı 2000 olacak şekilde eğitilmiştir. Eğitim verileri çıktıları listelenmiştir (Tablo 13).

Ortalama karesel hata öğrenme ve ortalama karesel hata doğrulama arasındaki fark ile ortalama karesel hata test değerleri bakımından elde edilen değerler birbirine çok yakın bulunmuştur. MinPoints parametresi için değerler çok fark etmese de, 1 gizli katmana sahip çok katmanlı perceptron modeli en uygun model olarak seçilmiştir (Tablo 14). MinPoints parametresi için en solda lineer regresyon, ortada 1 katmana sahip çok katmanlı perceptron modeli ve en sağda 2 katmana sahip çok katmanlı perceptron modeli ortalama karesel hata grafiği ve 3 boyutlu düzlemde elde edilen düzlemlerden bir kesit gösterilmiştir (Şekil 10). Doğrusal olmayan çözümlerden, 1 katmana sahip çok katmanlı perceptron modelinin daha kapsayıcı olduğu görülmektedir. Tablo 15 üzerinde kullanılan veriler başlığı altında 2 hafta boyunca toplanan konum verilerinin hangi haftalara ait olduğu yer almaktadır. Veri gönderme sıklığı ve veri paketi sayısı çok katmanlı algıyıcılar için kullanılan özelliklerdir

**Tablo 10.** Epsilon 2 parametresi için model sonuçları (Model results for the Epsilon 2 parameter)

Model	Dönem	Öğrenme Hızı	Ortalama Karesel Hata Öğrenme	Ortalama Karesel Hata Test	Ortalama Karesel Hata Doğrulama
Lineer Regresyon	2000	0.01	24.5309	16.8554	29.7313
Lineer Regresyon	2000	0.001	40.1407	31.8955	44.4264
Lineer Regresyon	2000	0.0001	2158.7526	2286.1645	2494.3520
1 Gizli Katman	2000	0.01	2.1966	6.0999	2.2535
1 Gizli Katman	2000	0.001	13.3279	9.9342	15.7525
1 Gizli Katman	2000	0.0001	153.5653	143.7384	188.4122
2 Gizli Katman	2000	0.01	0.1688	6.3210	0.1721
2 Gizli Katman	2000	0.001	2.5615	5.6403	2.6898
2 Gizli Katman	2000	0.0001	142.7105	132.5822	173.0727

**Tablo 11.** Epsilon 2 parametresine ilişkin uygun değerler (Appropriate values for the epsilon2 parameter)

Model	Dönem	Öğrenme Hızı	Ortalama Karesel Hata Öğrenme	Ortalama Karesel Hata Test	Ortalama Karesel Hata Doğrulama
Lineer Regresyon	650	0.01	24.6302	16.8076	29.9718
1 Gizli Katman	2000	0.01	2.1966	6.0999	2.2535
2 Gizli Katman	2000	0.01	0.1688	6.3210	0.1721

**Tablo 12.** Etiketli Epsilon2 parametresi ile modellerden elde edilen sonuçların karşılaştırması (Comparison of the results obtained from the models with the labeled Epsilon2 parameter)

Kullanılan Veriler	Veri Gönderme Sıklığı	Veri Paketi Sayısı	Etiketli Veriler Epsilon2	Doğrusal Regresyon	1 Gizli Katmana Sahip MLP	2 Gizli Katmana Sahip MLP
1. hafta ve 2. hafta	5	2760	20	20	22	19
1. hafta ve 2. hafta	10	1446	35	34	34	37
2. hafta ve 3. hafta	5	2747	20	20	22	19
2. hafta ve 3. hafta	10	1436	35	34	35	37
2. hafta ve 4. hafta	15	1000	50	45	45	49
2. hafta ve 4. hafta	20	837	60	55	54	56
3. hafta ve 4. hafta	15	989	50	45	45	50
3. hafta ve 4. hafta	20	828	60	55	54	56
3. hafta ve 4. hafta	25	692	65	65	64	63

(features). Etiketli veriler MinPoints başlığı altında ise daha önceden ST-DBSCAN algoritması için etiketlenmiş veri (labeled data) yer almaktadır. Doğrusal regresyon, 1 gizli katman sahip MLP ve 2 gizli katmana sahip MLP başlıkları altında ise yapay sinir ağlarına girdi verileri verildiğinde çıktı olarak elde edilen değerler yer almaktadır. Çalışma kapsamında, ST-DBSCAN girdi parametrelerini tahmin edecek modeller seçildikten sonra bu modellerin konum rutinlerini bulması beklenmektedir. Takip cihazının veri gönderme sıklığı ve toplam gönderilen veri paketi sayısı, ST-DBSCAN algoritmasının girdisi olan Epsilon1, Epsilon2 ve

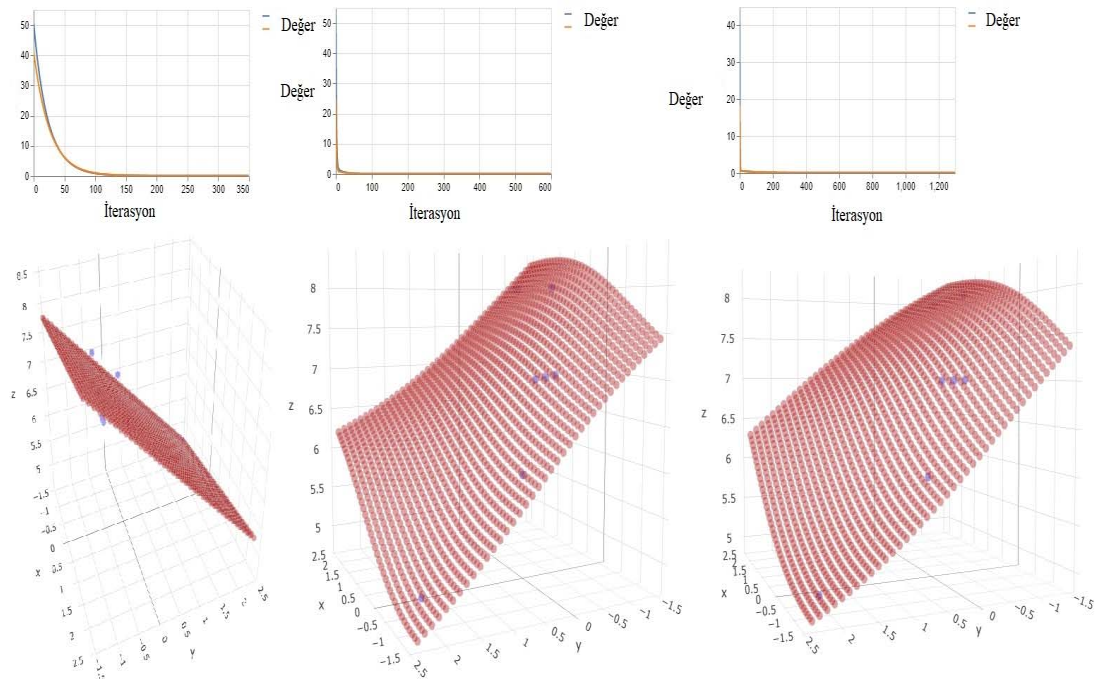
MinPoints parametrelerini tahmin için kullanılmıştır. Takip cihazından alınan konum ve zaman verileri ise ST-DBSCAN algoritmasının bir diğer girdisi olarak belirlenmiştir. Şekil 11'de gösterilen bir dizi yöntem uygulanarak, çoğu takip cihazında ortak olan veriler kullanılarak, 2 haftalık eğitim verisi ile takip edilen nesneye ait rutinlerin elde edilebileceği görülmektedir. Burada elde edilen rutin bilgileri içerisinde kalan bütün konum ve zaman verileri, takip edilen nesneye ait normal veriler olarak gösterilebilir. Bu rutin verilerin desenine uymayan bir veri ile karşılaşıldığında, bu veri anormal bir davranış ya da anormallik olarak işaretlenebilir.

**Tablo 13.** MinPoints parametresi için model sonuçları (Model results for the MinPoints parameter)

Model	Dönem	Öğrenme Hızı	Ortalama Karesel Hata Öğrenme	Ortalama Karesel Hata Test	Ortalama Karesel Hata Doğrulama
Lineer Regresyon	2000	0.01	0.0726	0.0756	0.0873
Lineer Regresyon	2000	0.001	0.0888	0.0932	0.1092
Lineer Regresyon	2000	0.0001	625315	763753.125	586397.5
1 Gizli Katman	2000	0.01	0.0553	0.0578	0.0677
1 Gizli Katman	2000	0.001	0.0754	0.0767	0.0826
1 Gizli Katman	2000	0.0001	49155.6367	52632.2031	19473.8339
2 Gizli Katman	2000	0.01	0.0548	0.0571	0.0667
2 Gizli Katman	2000	0.001	0.1318	0.1387	0.1631
2 Gizli Katman	2000	0.0001	124434.0390	160339.9062	101014.5625

**Tablo 14.** MinPoints parametresine ilişkin uygun değerler (Appropriate values for the MinPoints parameter)

Model	Dönem	Öğrenme Hızı	Ortalama Karesel Hata Öğrenme	Ortalama Karesel Hata Test	Ortalama Karesel Hata Doğrulama
Lineer Regresyon	350	0.01	0.0779	0.0800	0.0895
1 Gizli Katman	600	0.01	0.0531	0.0554	0.0648
2 Gizli Katman	1300	0.01	0.0554	0.0577	0.0672

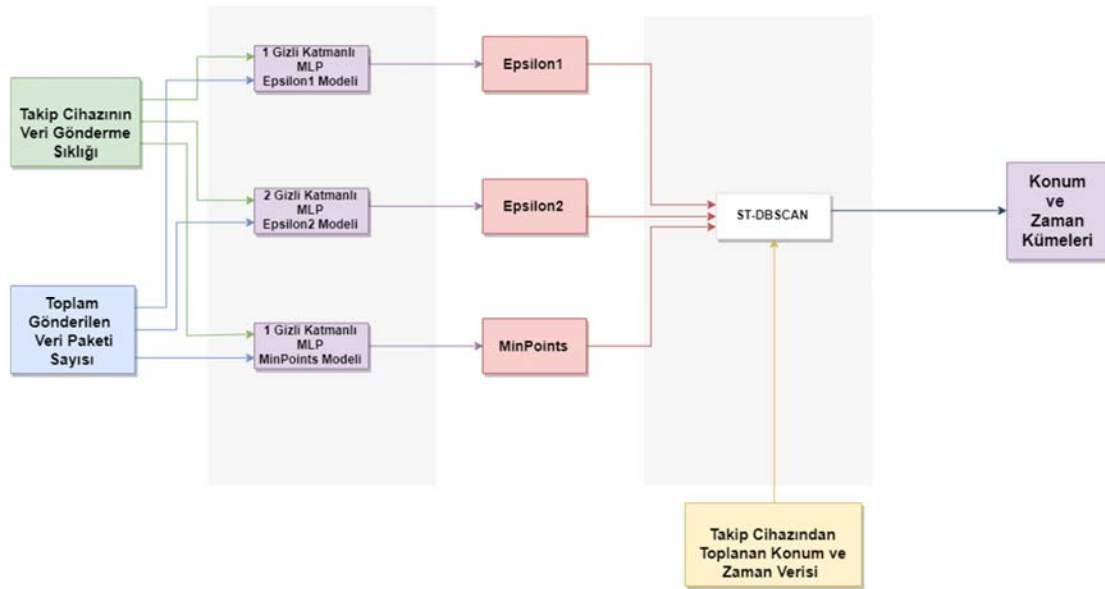
**Şekil 10.** MinPoints parametresi için ortalama karesel hata grafiği (Average square error graph for MinPoints parameter)

Normal konum davranışları bilinen takip nesnesinin, anlık olarak gelen konum verileri bu konum-zaman örüntüleri ile

karşılaştırılabilir ve normal davranışlar dışında kalan kısımlar var ise anormallik tespiti yapılabilir.

**Tablo 15.** Etiketli MinPoints parametresi ile modellerden elde edilen sonuçların karşılaştırması  
(Comparison of the results obtained from the models with the labeled MinPoints parameter)

Kullanılan Veriler	Veri Gönderme Sıklığı	Veri Paketi Sayısı	Etiketli Veriler MinPoints	Doğrusal Regresyon	1 Gizli Katmana Sahip MLP	2 Gizli Katmana Sahip MLP
1. hafta ve 2. hafta	5	2760	8	8.08	8.11	7.52
1. hafta ve 2. hafta	10	1446	8	7.59	7.68	7.26
2. hafta ve 3. hafta	5	2747	8	8.08	8.11	7.52
2. hafta ve 3. hafta	10	1436	8	7.58	7.68	7.26
2. hafta ve 4. hafta	15	1000	7	7.25	7.24	7.04
2. hafta ve 4. hafta	20	837	7	6.98	6.90	6.89
3. hafta ve 4. hafta	15	989	7	7.25	7.23	7.04
3. hafta ve 4. hafta	20	828	7	6.97	6.89	6.89
3. hafta ve 4. hafta	25	692	7	6.70	6.57	6.74

**Şekil 11.** Uygulanan yöntemlerin genel görünümü (General view of applied methods)

## 5. SONUÇLAR (CONCLUSIONS)

Bu çalışma sonucunda ST-DBSCAN algoritması ile konum ve zaman verilerine ilişkin rutinlerin tespit edilebileceği görülmüştür. Ayrıca, çalışma kapsamında kullanılan takip cihazı gibi farklı veri gönderme sıklığına ayarlanabilen cihazlardan elde edilen verilerde, ST-DBSCAN algoritmasının kullanılabilmesi için girdi parametrelerini tahmin edebilecek bir model geliştirilebileceği görülmüştür. Bu çalışma ile Eps ve MinPts girdi parametrelerini belirlemek için sezgisel yöntemlerin kullanılabilmesi ve çok katmanlı perceptron uygulamalarının doğrusal olmayan

verilerin üzerinde lineer regresyon uygulamasına göre daha verimli çalıştığı ve daha düzgün tahminlerde bulunduğu görülmüştür. Ayrıca, dışardan herhangi bir müdahaleye gerek kalmadan, takip edilen nesnenin rutinlerini ortaya çıkarabilecek bir yöntemler dizisi oluşturulmuştur. Bu çalışma kapsamında, zaman ve koordinat verisi olarak adlandırılan nesneye ait zamansal verilerden kümeleme algoritmaları yardımı ile benzer olanlar gruplandırılmıştır. Normal olarak nitelendirilen kümeler dışında bir veri oluştuğunda ise anormallik olarak tanımlanmıştır. Üzerinde çalışılan veri yapısı tüm takip cihazları için ortak olduğu için, bu çalışma kapsamında önerilen yöntem çok farklı alanlarda



kullanılabilir. Ayrıca, önerilen yöntem farklı gruplar için (çocuk, yaşlı, hasta vb.) erken uyarı sistemi olarak kullanılabilir. Gelecek çalışmalar kapsamında, derin öğrenme yaklaşımları kullanılarak çalışmada kullanılan verilerle konum kümeleri tahmin edilebilir. Çalışma sonucunda önerilen anormallik tespiti kısmı daha büyük veri kümelerine uygulanıp anlık olarak tespiti kontrol edilebilir. Bunun için tespit edilen kümelere ilişkin olarak kümelerin ne kadar dışındaki davranışların anormallik kapsamına gireceği konusunda bir öneri geliştirilebilir.

#### KAYNAKLAR (REFERENCES)

1. Topuz M.D., Makine Öğrenmesi Algoritmaları ve Anomali Tespiti, Yüksek Lisans Tezi, Bahçeşehir Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, 2014.
2. Bilgin T.T., Çamurcu Y., DBSCAN, OPTICS ve K-Means Kümeleme Algoritmalarının Uygulamalı Karşılaştırılması, *Politeknik Dergisi*, 8 (2), 139-145, 2005.
3. Birant D., Kut A., ST-DBSCAN: An algorithm for clustering spatial-temporal data, *Data and Knowledge Engineering*, 60, 208-221, 2007.
4. Alpaydın E., Yapay Öğrenme (çev. E. Alpaydın), Boğaziçi Üniversitesi Yayınevi, 2018.
5. Duan L., Xu L., Guo F., Lee J., Yan B., A local-density based spatial clustering algorithm with noise, *Information Systems*, 32 (7), 978-986, 2007.
6. Chandola V., Banerjee A., & Kumar V., Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41 (3), 15, 2009.
7. Kang M., Machine Learning: Anomaly Detection. In M. G. Pecht and M. Kang (Eds.), *Prognostics and Health Management of Electronics: Fundamentals, Machine Learning, and the Internet of Things*. John Wiley and Sons, NJ, 131-162, 2018.
8. Hong Z., Chen Y., and Mahmassani H., Recognizing Network Trip Patterns Using a Spatio-Temporal Vehicle Trajectory Clustering Algorithm, *IEEE Transactions on Intelligent Transportation Systems*, 2548 – 2557, 2017.
9. Yuan J., Zheng Y., Zhang C., Xie W., Xie X., Sun G., and Huang Y., T-Drive: Driving Directions Based on Taxi Trajectories, *Data and Knowledge Engineering*, 87, 357-373, 2013.
10. Lee J., and Whang K., Trajectory Clustering: A Partition-and-Group Framework, *SIGMOD International Conference on Management of Data*, 593-604, 2007.
11. Won J., Kim S., Baek J., and Lee J., Trajectory Clustering in Road Network Environment, *IEEE Symposium on Computational Intelligence and Data Mining*, 299-305, 2009.
12. Zhang D., Lee K., and Lee I., Mining Medical Periodic Patterns from Spatio-Temporal Trajectories. *Lecture Notes in Computer Science*, 123–133, 2018.
13. Gong S., Cartledge J., Bai R., Yue Y., Li Q., and Qiu G., Extracting Activity Patterns From Taxi Trajectory Data: A Two-Layer Framework Using Spatio-Temporal Clustering, *Bayesian Probability and Monte Carlo Simulation. International Journal of Geographical Information Science*, 1–25, 2019.
14. Wang Z., Hu J., and Fan Q., Extracting the Main Routes and Speed Profiles Between Two Locations from Massive Uncertain Historical Trajectories, *2018 International Symposium in Sensing and Instrumentation in IoT Era (ISSI)*, 2018.
15. Wai K. P., and Nwe Aung T., Distance-based Clustering of Moving Objects' Trajectories from Spatiotemporal Big Data\*, *2018 IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS)*, 2018.
16. Chimwayi K. B., and Anuradha J., Clustering West Nile Virus Spatio-temporal Data Using ST-DBSCAN, *Procedia Computer Science*, 132, 1218–1227, 2018.
17. Trisminingsih, R., & Shaztika, S. S., ST-DBSCAN clustering module in SpagoBI for hotspots distribution in Indonesia. In *2016 3rd International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, 327-330, IEEE, 2016.
18. Wang S., Cai T., Eick C.F., New spatiotemporal clustering algorithms and their applications to ozone pollution. In: *Proceedings of the 2013 IEEE 13th International Conference on Data Mining Workshops, ICDMW 2013*, IEEE Computer Society, Washington, DC, 1061–1068, 2013.
19. Joshi D., Samal A., Soh L.K., Spatio-temporal polygonal clustering with space and time as first-class citizens, *Geoinformatica*, 17 (2), 387–412, 2013.
20. Wang M., Wang A., Li A., Mining spatial-temporal clusters from geo-databases. In: Li, X., Za'iane, O.R., Li, Z. (eds.) *ADMA 2006. LNCS (LNAI)*, Springer, Heidelberg, 4093, 263–270, 2006.
21. Chen T., Bowers K., Cheng T., Zhang Y., & Chen P., Exploring the homogeneity of theft offenders in spatio-temporal crime hotspots. *Crime Science*, 9 (1), 1-13, 2020.
22. Wang W., Du S., Guo Z., Luo L., Polygonal clustering analysis using multilevel graph-partition, *Trans. GIS*, 19 (5), 716–736, 2015.
23. Zhang Y., Eick C.F., Novel clustering and analysis techniques for mining spatiotemporal data. In: *Proceedings of the 1st ACM SIGSPATIAL PhD Workshop, SIGSPATIAL PhD 2014*, 2:1–2:5. ACM, New York, 2014.
24. Damiani M.L., Issa H., Fotino G., Heurich M., Cagnacci F., Introducing 'presence' and 'stationarity index' to study partial migration patterns: an application of a spatio-temporal clustering technique, *Int. J. Geogr. Inf. Sci.*, 30 (5), 907–928, 2016.
25. Izakian H., Pedrycz W., Jamal I., Clustering spatiotemporal data: an augmented fuzzy c-means, *IEEE Trans. Fuzzy Syst.*, 21 (5), 855–868, 2013.
26. Izakian H., Pedrycz W., Anomaly detection and characterization in spatial time series data: a cluster-centric approach, *IEEE Trans. Fuzzy Syst.*, 22 (6), 1612–1624, 2014.
27. Lai W., Zhou M., Hu F., Bian K., & Song Q., A new DBSCAN parameters determination method based on

- improved MVO. *IEEE Access*, 7, 104085-104095, 2019.
28. Iyengar V.S., On detecting space-time clusters. In: *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 587–592. ACM, 2004.
  29. Schubert E., Zimek A., Kriegel H.P., Local outlier detection reconsidered: a generalized view on locality with applications to spatial, video, and network outlier detection. *Data Min. Knowl. Disc.* 28 (1), 190–237, 2014.
  30. Mohan P., Shekhar S., Shine J.A., Rogers J.P., Cascading spatio-temporal pattern discovery, *IEEE Trans. Knowl. Data Eng.*, 24 (11), 1977–1992, 2012.
  31. Shekhar S., Evans M.R., Kang J.M., Mohan P., Identifying patterns in spatial information: a survey of methods. *Wiley Interdisc. Rev.: Data Mining Knowl. Discov.*, 1 (3), 193–214, 2011.
  32. Nanni M., Pedreschi D., Time-focused clustering of trajectories of moving objects, *J. Intell. Inf. Syst.*, 27 (3), 267–289, 2006.
  33. Li Z., Spatiotemporal pattern mining: algorithms and applications. In: Aggarwal, C.C., Han, J. (eds.) *Frequent Pattern Mining*, Springer, Cham, 283–306, 2014.
  34. Palma A.T., Bogorny V., Kuijpers B., Alvares L.O., A clustering-based approach for discovering interesting places in trajectories. In: *Proceedings of the 2008 ACM Symposium on Applied Computing, SAC 2008*, ACM, New York, 863–868, 2008.
  35. Gudmundsson J., van Kreveld M., Computing longest duration flocks in trajectory data. In: *Proceedings of the 14th Annual ACM International Symposium on Advances in Geographic Information Systems, GIS 2006*, ACM, New York, 35–42, 2006.
  36. Jeung H., Yiu M.L., Zhou X., Jensen C.S., Shen H.T., Discovery of convoys in trajectory databases, *Proc. VLDB Endow*, 1 (1), 1068–1080, 2008.
  37. Li Z., Ding B., Han J., Kays R., Swarm: mining relaxed temporal moving object clusters, *Proc. VLDB Endow*, 3 (1–2), 723–734, 2010.
  38. Scitovski R., Sabo K., DBSCAN-like clustering method for various data densities, *Pattern Analysis and Applications*, 1-14, 2019.
  39. Wang X., Zhang Y., Liu H., Wang Y., Wang L., & Yin B., An Improved Robust Principal Component Analysis Model for Anomalies Detection of Subway Passenger Flow, *Journal of Advanced Transportation*, 2018.
  40. Ester M., Kriegel H.-P., Sander J., Xu X., A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In E. Simoudis, J. Han, U. Fayyad (Eds.). *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. American Association for Artificial Intelligence, Portland, 226-231, 1996.
  41. Pasin O., Ankarali H., Usage of Kernel K-Means and DBSCAN Cluster Algorithms in Health Studies: An Application, *Clin Res Trials*, 1, doi: 10.15761/CRT.1000116, 2015.
  42. Çakıt E., Durgun B., & Cetik O., A neural network approach for assessing the relationship between grip strength and hand anthropometry, *Neural Network World*, 25 (6), 603, 2015.
  43. Çakıt E., Karwowski W., Estimating electromyography responses using an adaptive neuro-fuzzy inference system with subtractive clustering, *Human Factors and Ergonomics in Manufacturing & Service Industries*, 27 (4), 177-186, 2017.
  44. Başoğlu B., Bulut M., Development of a hybrid system based on neural networks and expert systems for shortterm electricity demand forecasting, *Journal of the Faculty of Engineering and Architecture of Gazi University*, 32 (2), 575-583, 2017.
  45. Yıldız O., Melanoma detection from dermoscopy images with deep learning methods: A comprehensive study, *Journal of the Faculty of Engineering and Architecture of Gazi University*, 34 (4), 2241-2260, 2019.
  46. Çakıt E., Karwowski W., Understanding the social and economic factors affecting adverse events in an active theater of war: a neural network approach. In *International Conference on Applied Human Factors and Ergonomics* Springer, Cham, 215-223, 2017.
  47. Choubin B., Khalighi-Sigaroodi S., Malekian A., & Kişi Ö., Multiple linear regression, multi-layer perceptron network and adaptive neuro-fuzzy inference system for forecasting precipitation based on large-scale climate signals, *Hydrological Sciences Journal*, 61 (6), 1001-1009, 2016.