Makale Türü: Araştırma Makalesi
Zeki Sistemler Teori ve Uygulamaları Dergisi 3(2) (2020): 60-68
DOI: 10.38016/jista.755481

Otistik Spektrum Bozukluğunun Makine Öğrenme Algoritmaları ile Tespiti

Sedat Metlek[1*], Kıyas Kayaalp[2]

[1]Burdur Mehmet Akif Ersoy Üniversitesi, Teknik Bilimler Meslek Yüksekokulu, Elektronik ve Otomasyon Bölümü

[2]Isparta Uygulamalı Bilimler Üniversitesi, Uluborlu Selahattin Karasoy Meslek Yüksekokulu, Bilgisayar Teknolojileri Bölümü

sedatmetlek@mehmetakif.edu.tr, kiyaskayaalp@isparta.edu.tr

Öz

Farklı etkileri bulunan Otistik Spektrum Bozukluğu (OSB) genel olarak sosyal ilişki ve bilişsel gelişimde gecikme ya da farklılaşma ile kendini gösteren ayrıca iletişim de sorunlara neden olan nöro-gelişimsel bir hastalıktır. Hastalığın, bireylerin gelişimine ve ileriki dönemlerdeki sosyal yaşantılarına olumsuz etkisini azaltmak için erken teşhis edilmesi oldukça önemlidir. Ancak OSB'nin erken yaşlarda tespit edilebilmesi tecrübe ve uzmanlık gerektirmektedir. Son yıllarda yapılan araştırmalarda Dünya genelinde ve Türkiye'de OSB vakalarında ciddi bir artışın olduğu gözlenmektedir. Böyle bir artışta her geçen gün erken teşhis için etkili ve kolay uygulanabilir teşhis yöntemlerine olan ihtiyacı artırmaktadır. Özellikle 12-36 ay arasındaki çocuklara OSB teşhisi konulabilmesi için yardımcı karar destek sistemlerinin geliştirilmesi hayati önem arz etmektedir.

Gerçekleştirilen çalışmada, 12-36 ay arasındaki çocuklara uzman sağlık personeli ve ailelerin yüksek doğrulukta OSB teşhisi koyabilmelerine yardımcı olabilecek bir karar destek yazılımı geliştirilmiştir. Yazılım geliştirme aşamasında gözetimli ve gözetimsiz olmak üzere altı farklı makine öğrenme algoritması test edilmiştir. Yapılan testler sonucunda gözetimli öğrenme algoritmalarının, gözetimsiz öğrenme algoritmalarına göre daha başarılı sonuçlar verdiği tespit edilmiştir. Kullanılan gözetimli öğrenme algoritmalarında destek vektör makineleri ile yapılan sınıflandırma işleminde %100 sınıflandırma başarım oranı elde edilmiştir.

Anahtar kelimeler: Otistik spektrum bozukluğu, gözetimli makine öğrenmesi, gözetimsiz makine öğrenmesi, yardımcı karar destek sistemi.

Detection of Autistic Spectrum Disorder with Machine Learning Algorithms

Abstract

Autistic Spectrum Disorder (ASD), which has different effects, is a neurodevelopmental disease that generally manifests with delay or differentiation in social relationship and cognitive development, and also causes problems in communication. It is very important to diagnose the disease early to reduce the negative impact on the development of individuals and their social life in the future. However, it has been require experience and expertise to detect ASD at an early age. In researches conducted in recent years, it is observed that there is a significant increase in ASD cases in the world and Turkey. With such an increase, the need for effective and easily applicable diagnostic methods for early diagnosis increases day by day. It is vital to develop auxiliary decision support systems, especially for children between 12-36 months to be diagnose with ASD.

In the study, a decision support software was developed that could help specialist medical staff and families diagnose OSB with high accuracy in children between 12 and 36 months of age. Six different machine learning algorithms, both supervised and unsupervised, were tested during the software development phase. As a result of the tests, it has been determined that supervised learning algorithms give more successful results than unsupervised learning algorithms. In the supervised learning algorithms used, 100% classification success rate was obtained in the classification process with the support vector machines.

Keywords: Autistic Spectrum Disorder, supervised machine learning, unsupervised machine learning, auxiliary decision support system.

* Sorumlu yazar: Sedat METLEK
E-posta adresi: sedatmetlek@mehmetakif.edu.tr

Alındı : 20 Haziran 2020
Revizyon : 27 Ağustos 2020
Kabul : 28 Ağustos 2020

1. Giriş (Introduction)

Otistik Spektrum Bozukluğu (OSB), doğuştan yada erken yaşlarda ortaya çıkan bir nöro-gelişimsel farklılıktır. Hastanın ilerleyen dönemlerdeki yaşayacağı olumsuzluklar, erken tanı ve eğitim ile önemli ölçüde azaltılabilmektedir. Fakat, OSB teşhisi için tanı koyma süreçleri uzun ve tecrübeye dayalı işlemlerdir (Kayaalp and Metlek, 2020; Metlek, 2018). Yapılan araştırmalara bakıldığında Dünya genelinde OSB vakalarında artış olduğu görülmektedir. Bu nedenle etkili ve kolay uygulanan teşhis yöntemlerinin geliştirilmesine acil ihtiyaç vardır. Bu teşhis yöntemlerinin içerisinde, sağlık çalışanlarına yardımcı olmak ve hastalara klinik tanı koymak süresini azaltmak için yardımcı karar destek yazılımlarının geliştirilmesine de ihtiyaç vardır.

Literatürde konu ile ilgili yapılan çalışmalar incelendiğinde, Thabtah ve Peebles'in 2019 yılındaki çalışmasında çocuk, ergen ve yetişkinlerde OSB teşhisinde kullanılabilecek bir yazılım geliştirdiği görülmektedir. Geliştirdikleri yazılımda kullandıkları yöntemler ile yetişkinler için ortalama %90 ile %95, çocuklar için %85 ile %90 ve ergenler için %65 ile %85 arasında teşhis başarısı elde etmişlerdir. Yapmış oldukları çalışmanın konusunda 12-36 ay arasındaki bebekler bulunmamaktadır (Thabtah ve Peebles, 2019).

Cho ve arkadaşları 2019 yılında, çocukların kısa doğal konuşmalarındaki akustik metin özelliklerinden yararlanarak OSB teşhisini gerçekleştirmişlerdir. Yaptıkları çalışmada gradient tabanlı bir algoritma kullanmışlardır. Kullandıkları verileri azaltmak içinde temel bileşen analizi yöntemini tercih etmişlerdir. Çalışmalarının sonucunda, OSB teşhisinde ortalama %76 başarı sağlamışlardır (Cho vd., 2019).

Eslami ve arkadaşları 2019 yılında fMRI verileri ile otomatik kodlayıcı ve tek katmanlı algılayıcı kullanarak hibrit bir öğrenme prosedürü tasarlamışlardır. Çalışmalarında %82 başarı oranı ile OSB teşhisini gerçekleştirmişlerdir (Eslami vd., 2019).

Shahamiri ve arkadaşı 2018 yılında hazırladıkları mobil bir uygulama ile kişilerin verdiği cevaplar üzerinden OSB teşhisini yapmışlardır (Shahamiri ve Thabtah, 2018). 2019 yılında Dawson ve arkadaşı ise, mobil cihazlar üzerinden alınan görüntüler ve makine öğrenmesi ile OSB teşhisinin gerçekleştirilebileceğini önermektedir (Dawson ve Sapiro, 2019).

Küçük çocuklarda özellikle OSB teşhisini koymak son derece güçtür. Bunun da en önemli nedeni; çocuklara sorulan sorulardan anlaşılır bir cevabın alınması zordur. Bu nedenle çocukların kendi aile bireyleri yada bakıcılarından alınan cevaplara göre bu teşhis konulmaya çalışılmaktadır.

Çalışma ile ilk olarak 12-36 ay arasındaki çocuklarda OSB teşhisinin konulabilmesi amaçlanmıştır. Bu amaç doğrultusunda literatüre önemli bir katkı sağlanmıştır. Bu katkıyı sağlayabilmek için ilk etapta OSB teşhisi konulan kişilere ait davranış özelliklerini içeren veri setlerine ihtiyaç vardır. Bazı Avrupa ülkeleri hariç birçok ülkede konu ile ilgili geniş kapsamlı bir veri seti bulunmamaktadır.

Halbuki bu tür veri setleri, OSB teşhis sürecini kısaltmak için geliştirilecek yazılımların verimliliğini, duyarlılığını, özgüllüğünü ve başarım oranını arttırarak daha detaylı analizler yapılmasını sağlamaktadır. Dünya geneline bakıldığında, uluslararası düzeyde kaynak veri sağlayan Manukau Teknoloji Enstitüsünün OSB ile ilgilenen araştırmacılara sunmuş olduğu veri seti bu çalışmada kullanılmıştır (Thabtah, 2018). Gerçekleştirilen çalışmada bir kişi için 16 farklı özellik içeren bu veri seti kullanılarak, makine öğrenme algoritmaları ile hastalara hızlı ve güvenilir teşhis koymak için Matlab ortamında bir yazılım geliştirilmiştir. Geliştirilen yazılım Intel(R) Core(TM) i7-4702MQ 2.2GHz işlemci, 16 GB Ram, NVIDIA GeForce GT 740M 2 GB ekran kartı bulunan bir bilgisayarda geliştirilmiştir.

Çalışmanın literatüre önemli bir diğer katkısı da Manukau Teknoloji Enstitüsünün sunmuş olduğu veri seti üzerinde, gözetimli ve gözetimsiz altı farklı makine öğrenme algoritmasının, OSB teşhisindeki başarısının ayrıntılı olarak tespit edilmesidir.

2. Yöntem (Methodology)

Literatürde bulunan makine öğrenme algoritmaları genel olarak gözetimli ve gözetimsiz olmak üzere ikiye ayrılmaktadır. Gözetimli öğrenme algoritmaları olarak da bilinen, bilgi tabanlı öğrenme algoritmalarında veriler, eğitim ve test olarak ikiye ayrılmakta ve eğitim verilerinden anlamlı bilgilerin elde edilmesi amaçlanmaktadır. Verilerin eğitim ve test olarak ayrılmasının yanında, bazı durumlarda hangi özelliklerinin anlamlı olabileceği konusunda uzman görüşü de gerekebilmektedir. Bu da zaman ve maliyet açısından pahalı sayılabilecek bir yöntem olmasına neden olmaktadır.

Literatürde ki bir diğer makine öğrenme algoritması da gözetimsiz makine öğrenme algoritmasıdır. Bu yaklaşımda gözetim işlemi olmadığından dolayı zaman ve işlem maliyeti açısından diğer yöntemden daha ucuz bir öğrenme modelidir.

Gözetimli ve gözetimsiz öğrenme modelleri arasındaki en önemli farklılık; gözetimli öğrenme modelinde, eğitim setinde bulunan etiketlenmiş veriler ile bir fonksiyonun üretilmesidir. Gözetimsiz öğrenme modelinde ise veri setini oluşturan etiketsiz verilerden oluşacak sınıfları tahmin etmek için bir fonksiyonun üretilmesidir (Çürükoğlu, 2019b).

2.1. Gözetimli makine öğrenmesi (Supervised machine learning)

Bu makine öğrenmesi yönteminde, sınıflandırma öncesinde genellikle sınıf bilgilerinin çıkarılabileceği bir veri seti gerekmektedir. Bu veri setinden elde edilen değerler, test verilerinde kullanılarak öğrenme işlemi gerçekleştirilmektedir. Genel olarak, literatürde

kullanılan gözetimli makine öğrenme algoritmaları (Ülgen, 2017;Çürükoğlu, 2019a):

- K-En Yakın Komşuluk,
- Destek Vektör Makinaları,
- Karar Ağaçlarıdır.

2.1.1.K-En yakın komşuluk (K-Nearest neighbors (K-NN))

K-NN bilinen en eski ve basit sınıflandırma algoritmalarından birisidir. Algoritmanın temeli, örnek veri ile k adet komşu arasındaki mesafenin ölçülmesine dayanmaktadır. Aradaki mesafe hangi mesafe ile daha az ise örnek o sınıfa dahil edilmektedir (Cover ve Hart, 1967). K-NN sınıflandırma algoritmasında k değeri, komşular arasındaki uzaklık ve ağırlıklandırma ölçütleri sınıflandırma performansını doğrudan etkilemektedir.

K-NN algoritması gözetimli bir algoritma olması nedeniyle, veriler ilk olarak eğitim ve test olarak ikiye ayrılmaktadır. K-NN algoritmasına sınıflandırma için yeni bir değer geldiğinde, bu değerin hangi sınıfa ait olduğu, eğitim setindeki örneklere olan uzaklığına bakılarak karar verilir. Burada kaç adet komşuya bakılacağı k komşuluk katsayısına göre karar verilir. Sonraki aşamada da k adet komşuya olan uzaklıkların çoğunluk oylaması yapılır ve gelen verinin ait olduğu sınıf belirlenir. Gelen veri ile etrafındaki k adet komşusu ile aradaki uzaklığı ölçmek için kullanılan bazı fonksiyonlar aşağıda gösterilmiştir.

Minkowski

$$\left(\sum_{i=1}^{k}|x_i - y_i|^p\right)^{1/p} \quad (1)$$

Öklid

$$\left(\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}\right) \quad (2)$$

Manhattan

$$\left(\sum_{i=1}^{k}|x_i - y_i|\right) \quad (3)$$

Chebyschev

$$lim_{p\to\infty}\left(\sum_{i=1}^{k}|x_i - y_i|^p\right)^{1/p} = max_{i=1}^{k}|x_i - y_i| \quad (4)$$

Denklem 1, 2, 3, 4'de kullanılan x_i değeri, eğitim örnekleri arasındaki i adet örneği, y_i değeri ise sınıflandırma için gelen i adet yeni veriyi ifade etmektedir. k değeri ise yeni gelen veri ile etrafındaki kaç adet komşu ile arasındaki mesafenin ölçüleceğini belirten parametredir (Taşcı ve Onan, 2016).

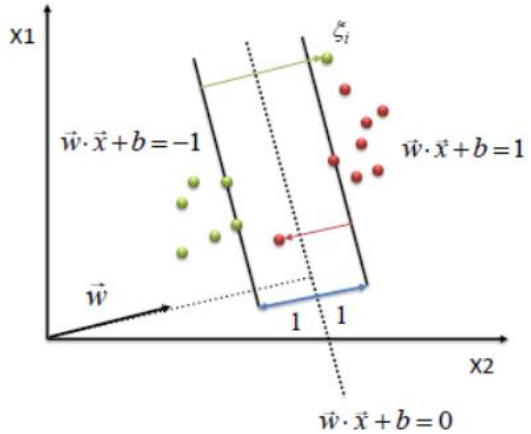
2.1.2. Destek vektör makineleri (Support vector machine (SVM))

SVM, istatistiksel teoriler üzerine dayanan güçlü bir makine öğrenme algoritmasıdır. Geleneksel yapay zeka algoritmalarında öğrenme için çok sayıda eğitim verisi gerekmektedir. Bunun ile birlikte birçok sınıflandırma algoritmasında, düşük yakınsama oranı, yerel minimuma takılma ve ezberleme problemleriyle karşılaşılmaktadır (Lu vd., 2002). Fakat SVM'nin kararlı yapısı sayesinde bu tarz problemler sorun teşkil etmemektedir. Ayrıca çok boyutlu ve az sayıda veri içeren çalışmalarda SVM'ler başarılı sonuçlar vermektedir. Literatürde sınıflandırma uygulamalarında tercih edilen SVM modeli, destek vektör sınıflandırmasıdır. Regresyon çalışmalarında kullanılan SVM modeli ise destek vektör regresyonudur (Shen, Pei, ve Lee, 2004;Taburoğlu, 2019).

SVM'lerin amacı, sınıflandırmak istenilen sınıflar arasındaki en yüksek uzaklığa sahip ayırıcı düzlemi Şekil 1'de gösterildiği üzere bulmaktır. Denklem 5 ile bu ayırıcı düzlem formülüze edilmiştir.

$$f(x) = (w, x) + b \quad (5)$$

Denklem 5'deki w ağırlık katsayısını, b bias (fişekleme) değerini ve x eğitim için kullanılan veriyi ifade etmektedir.



Şekil 1. SVM yapısı (Structure of SVM) (Sayad, 2015)

SVM'deki eğitim verileri 1×N boyutlu bir matris ile ifade edilir. Veri kümesinde bulunan m sayıdaki veri y ϵ {+1,-1} kümesindeki değerlerden birisiyle eşleştirilir. Bu eşleşme işlemi için Denklem 6 kullanılır ve burada $\xi_i \geq 0$ şartının sağlanması gereklidir.

$$y_i[(w, x_i) + b] \geq 1 - \xi_i, i = 1, \ldots, m \quad (6)$$

Optimum ayırıcı düzlemin bulunması için Denklem 6'daki koşula bağlı olarak Denklem 7'deki amaç fonksiyonunun minimum değeri bulunmalıdır.

$$C\sum_{i=1}^{n}\xi_i+\frac{1}{2}\|w\|^2 \quad (7)$$

Denklem 6 formundaki koşullar kullanılarak Denklem 7 minimum yapılmaktadır. Denklem 7'deki C, kullanıcı tarafından tanımlanan, sınıflandırma doğruluğu ile ayırıcı düzlemin karmaşıklığı arasındaki dengeyi sağlayan sıfırdan büyük bir sayıdır, ξ ise esneklik katsayısıdır. Denklem 6 ve 7'de belirtilen problem, Lagrange yöntemiyle binary optimizasyon problemi yapısında tekrar düzenlenirse Denklem 8.1 elde edilir. Denklem 8.2'deki eşitlik ve eşitsizlik durumlarına göre Denklem 8.1'in en yüksek değeri elde edilmeye çalışılır. Denklem 8.1'deki değer, giriş verilerinin iç çarpımları ile doğrudan orantılıdır.

Maksimum

$$Q(\alpha)=\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,k=1}^{n}\alpha_i\,y_i\,\alpha_k\,y_k\,(x_i\,,x_k) \quad (8.1)$$

koşul,

$$\sum_{i=1}^{n}\alpha_i\,y_i=0,\qquad \alpha_i\geq 0,\forall i \quad (8.2)$$

Sıfırdan büyük α_i Lagrange çarpanları ile Denklem 9 elde edilir ve bu çarpanlara ait verilere destek vektör denilir. Veri kümesi içerisindeki ayırıcı düzlemi en iyi ifade eden veriler, destek vektör verilerdir.

$$f(x)=sign\left(\sum_{i=1}^{\#sv}\alpha_i y_i K(x\,,x_i)+b\right) \quad (9)$$

Optimizasyon probleminde yukarıda belirtilen SVM denklemleri kullanılarak optimum Lagrange çarpan değerleri elde edilir. Bu değerler kullanılarak da ayırıcı düzlem oluşturulur. Ayırıcı düzlem oluşturulurken sadece destek vektör değerleri alındığı için tüm verilere göre daha seyrek noktalardan oluşur. Bu aşamaya kadar belirtilen SVM sınıflandırıcısı lineer olarak ayrılabilecek çalışmalarda yüksek başarı oranları ile kullanılabilir. Fakat, lineer olmayan çalışmalarda ise yüksek bir başarım oranı hedeflenirse çekirdek fonksiyonlarının kullanılması gerekmektedir. Giriş değerlerinden, lineer olmayan veriler çekirdek fonksiyonu yardımı ile çok boyutlu lineer nitelik uzayına aktarılır. Bu aktarım işlemi Denklem 10'da belirtilen fonksiyon gibi çekirdek fonksiyonlar ile gerçekleştirilir.

$$K(x\,,x_i)=K(x_i,x)=\varphi(x)^T\varphi(x_i) \quad (10)$$

Çalışmada tercih edilen çekirdek fonksiyonu, Radial Base Fonksiyonudur (RBF). RBF çekirdeğinin içeriği Denklem 11'de gösterilmiştir.

$$K(x\,,x')=exp\left(\frac{-\|x-x'\|^2}{\sigma^2}\right) \quad (11)$$

Denklem 11'deki σ, genişliği ifade eden kullanıcı tanımlı sıfırdan büyük reel bir sayıdır.

2.1.3. Karar ağaçları (Decision trees (DT))

İstatistiksel olarak anlamlı grupları bulan DT, sınıfları kolay ve anlaşılabilir ağaç diyagramları halinde ifade eden bir makine öğrenme yöntemidir (Dogan ve Ozdamar, 2003). Gözetimli bir öğrenme yöntemi olan DT'ler girdi ve çıktı kümelerinden oluşmaktadır. DT, sınıf çıktılarının durumu ile girdi değişkenleri arasındaki yapıyı keşfeder (Tan, Steinbach, ve Kumar, 2005).

DT yapılarında aynı sınıfa ait olan veriler yapraklarda bulunur (Huang, Lu, ve Ling, 2003). Bir DT kök, dal, yaprak ve bunlar arasındaki düğümlerden oluşan bir sisteme sahiptir. Bir dal üzerinde bulunan yaprakta, olası bütün çıktı sınıflarına karşılık gelen sonuç değerleri bulunmaktadır (Kantardzic, 2011). DT yapısında veri miktarının fazlalığı sonucu etkilemez. DT'nin yapısı için oluşturulan teknikler hesaplama yöntemi olarak ucuz ve hızlıdır (Tan, Steinbach, ve Kumar, 2016).

DT yapısında çok fazla sınıf bulunması durumunda, ağacın yapısı genişlemekte, bunun sonucunda düğüm sayısı artarken düğümlerdeki sınıf bilgisi azalmaktadır. Bu durumda da sistemin güvenirlilik oranı düşmektedir (Seidman, 2001).

2.2. Gözetimsiz makine öğrenmesi (Unsupervised machine learning)

Gözetimsiz makine öğrenme yaklaşımların genel amacı, veri setinde bulunan etiketsiz bilgiler arasındaki gizli ilişkilerin veya grupların ortaya çıkarılmasıdır. Literatürde kullanılan bazı temel gözetimsiz öğrenme algoritmaları aşağıda sunulmuştur (Yumuş, 2019);

- K-Ortalamalar,
- Temel Bileşen Analizi,
- Birliktelik Kuralları.

2.2.1. K-Ortalamalar algoritması (K-means algorithm)

K-Means algoritması, gözetimsiz bir makine öğrenme algoritmasıdır. K-Means ifadesindeki K değeri, küme sayısını ifade eder. Geliştiricinin dışarıdan mutlaka bu K değerini algoritmaya girmesi gerekmektedir. Bu durum bazı uygulamalarda dezavantaj olabilmektedir. Bu nedenle geliştirilmiş X-Means gibi benzer algoritmalar da bulunmaktadır. Algoritmanın sade bir çalışma şekli vardır.

Algoritmada Denklem 12'de ve Tablo 1'de gösterildiği üzere D ile ifade edilen veri kümesi, m-boyutlu reel bir vektördür. Denklem 13' deki x_1, x_2, …, x_m şeklinde ifade edilen değerler ise bir durum için elde edilen n boyutlu vektör tipindeki veri setidir. Bu veri setindeki her bir değer, bir olaydaki özellikleri ifade

etmektedir. K-means makine öğrenmesi, karesel hatayı en aza indirgemek için m adet veriyi K adet kümeye bölmeyi amaçlar.

$$D = (x_1, x_2, x_3, \dots, x_m) \quad (12)$$
$$x_i = (x_{i1}, x_{i2}, x_{i3}, \dots, x_{in}) \quad (13)$$

Tablo 1. D veri kümesi (D dataset)

		1. özellik	2. özellik	3. özellik	n. özellik
x_1	=>	x_{11}	x_{12}	……	x_{1n}
x_2	=>	x_{21}	x_{22}	……	x_{2n}
·	=>				
·					
·	=>				
x_m		x_{m1}	x_{m2}	……	x_{mn}

K değeri belirlendikten sonra, var olan değerler içerisinden rastgele K tane küme merkez noktası μ_j seçilir. Rastgele belirlenen merkez noktaları ile her bir veri için Denklem 14 ve 15'e göre veriler arasındaki uzaklık hesaplanarak veriyi, en yakın merkez noktasına göre bir kümeye dahil eder (Edureka, 2020).

$$C_j = Küme(x_i) = arg_j min\left\|x_i - \mu_j\right\|^2 \quad (14)$$

$$Değişim = \sum_{i=1}^{m}\left(x_j - c_j\right)^2 = \sum_{j=1}^{k}\sum_{i=1}^{\mu_j}\left(x_i - \mu_j\right)^2 \quad (15)$$

Sonraki aşamada her küme için yeni bir merkez noktası hesaplanır ve yeni merkez noktalarına göre kümeleme işlemi tekrar edilir. Bu işlem, sistemdeki değişim kararlı hale gelinceye kadar tekrar edilir.

K-Means algoritmasındaki genel amaç, elde edilen kümelerin, küme içi benzerliklerinin en yüksek ve kümeler arası benzerliklerinin en az olmasını sağlamaktır. Çalışma yönteminde, Denklem 2 ve 3'deki formüller temel alınarak kümeler arasında ki mesafeler hesaplanmaktadır.

2.2.2. Temel bileşen analizi (Principal Component Analysis (PCA))

PCA yöntemi, çok boyutlu bir veri setindeki verileri, temel özelliklerinden tespit ederek daha az sayıda değişkenle ifade edilmesi için geliştirilen yöntemlerden birisidir.

Birçok alanda sıklıkla tercih edilen PCA kısaca bir lineer boyut azaltma yöntemidir. PCA, veri setinin birbiri ile ilişkili değişkenlerini, ortogonal dönüşüm yöntemini esas alarak birbiri ile ilişkisiz değişkenlere dönüştürmeyi amaçlamaktadır. Bu işlem esnasında da boyut azaltma işlemi gerçekleştirilmektedir.

PCA yöntemi, m boyutlu n örnekli bir veri seti ($X = [x_1, x_2, x_3, \dots\dots, x_n]$) için, kovaryans matrisinin toplamının (Σ) bulunması ile başlar. Denklem 16' daki x_i, i. örneğin özniteliklerini ifade etmektedir. $\bar{x}$ terimi ise veri nesnelerinin ortalama değerini ifade etmektedir.

$$\sum = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})\,(x_i - \bar{x})^T = \frac{1}{n}XX^T \quad (16)$$

Denklem 16'da kullanılan $\bar{x}$ nin içeriği Denklem 17'de verilmiştir.

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n}(x_i) \quad (17)$$

Kovaryans matrisinin öz değerleri ve öz vektörleri, öz değer-öz vektör ayrıştırma yöntemi ile bulunur (Yıldız ve Sevim, 2016).

2.2.3. Birliktelik kuralları (Association rules)

Birliktelik kuralları veri setindeki bir dizi verinin diğer verilerle olan bağlantısını ortaya koyan bir kümeleme yöntemidir. Bu yöntemin matematiksel analizi aşağıda detaylı olarak verilmiştir (Agrawal ve Srikant, 1994;Eker, Oktaş, ve Kayhan, 2015).

Veri kümesi X ile gösterilirse, veri kümesindeki her bir veri seti $\{X_1, X_2, X_3, \dots, X_m\}$ şeklinde bir dizi ile ifade edilir. Veri setindeki işlemlerde $Y = \{y_1, y_2, y_3, \dots, y_k\}$ şeklinde gösterilirse, y_k' nın alacağı değer 0 veya 1'dir. Eğer $y_k = 0$ ise X_k veri setinin işleme alınmadığını, $y_k = 1$ ise X_k veri setinin işleme alındığını ifade etmektedir. Her bir işlem için veri setinde ayrı ayrı kayıtlar vardır. $Z \subseteq X$ için Z' deki her bir X_k'ya karşılık gelen bir y_k değeri bulunmaktadır ve $y_k = 1$'dir. Bu birliktelik durumu aşağıdaki şekilde ifade edilmektedir.

$Z \Rightarrow X_j$, Z, X 'in bir alt kümesidir. X_j ise X'in içerisinde herhangi bir elemandır. Bu eleman Z'nin içerisinde değildir. $Z \Rightarrow X_j$ kuralının Y için uygun olduğunun söylenebilmesi için belli bir güven seviyesinden söz etmek gerekmektedir. Yani, Y içindeki tüm X'lerin ne kadarının X_k'yı sağladığı %c değeriyle ifade edilmelidir. Bu durumda, birliktelik kuralını $0 \leq c \leq 1$ güven seviyesiyle birlikte $Z \Rightarrow X_j$, $Z \Rightarrow X_j|c$ şeklinde ifade edilebilir. Güven seviyesi, kuralın gücünü ifade etmektedir. Buna ek olarak, kuralın destek seviyesinden de söz edilir. Destek seviyesi ise Y içindeki işlemlerin ne kadarının Z'yi sağladığıdır.

3. Uygulama (Application)

Gerçekleştirilen uygulamada Manukau Teknoloji Enstitüsünün bebeklerdeki (12-36 ay) otistik spektrum bozukluğunun tespiti için sunmuş olduğu, 16 farklı özellik içeren 1054 kayıt kullanılmıştır.

Veri setinde bulunan özellikler, bebeklerde otistik spektrum bozukluğunun bulunup bulunmadığını tespit etmek için kullanılmıştır. Tablo 2'de bu tespit için kullanılan veri setindeki 16 giriş değeri sunulmuştur.

Tablo 2. Veri setindeki bilgiler (Information on dataset) (Shahamiri ve Fadi, 2018)

No	Değer
1	Çocuğun kendi ismi ile hitap edildiğinde size bakma durumu
2	Çocuk ile göz teması kurma durumu
3	Çocuğun bir şey istediğinde onu işaret etme durumu
4	Çocuğun ilginç bulduğu bir şeyi karşısındakine işaret etme durumu
5	Çocuğun taklit yapmayı gerektiren oyunları oynama durumu
6	Çocuğun karşısındakinin baktığı yeri gözleri ile takip etme durumu
7	Ailedeki birinin mutsuz olması durumunda çocuğun ilgili kişiye tepki durumu
8	Çocuğun konuştuğu ilk sözcük nasıl tanımlanır
9	Çocuğun iletişim kurmak için basit beden hareketlerini kullanma durumu
10	Çocuğun sebepsiz yere bir yere odaklanma durumu
11	Yaş (Ay)
12	Cinsiyet
13	Etnik köken
14	Sarılık hastalığını geçirme durumu
15	Ailede otistik spektrum bozukluğu var mı
16	Verileri kimin girdiği (Aile/Başkası)

Çalışmada ilk olarak gözetimli makine öğrenme modelleri anlatılan sıra ile uygulanmıştır. Çalışmada konu ile ilgili toplam 1054 örnek bulunmaktadır. Bu örneklerin 844'ü eğitim (%80), 210'u da test (%20) için kullanılmıştır.

3.1. K-En yakın komşuluk (K-Nearest neighbors)

K-en yakın komşu algoritmasında, öznitelik uzayındaki ele alınan bir noktanın etrafında bulunan komşuları ile arasındaki mesafeye bakılarak kümeleme işlemi yapılmaktadır (Alpaydin, 2020).

Tablo 3. K-en yakın komşuluk başarım sonuçları (K-nearest neighborhood performance results)

Uzaklık fonksiyonu	Komşuluk sayısı (K)	Başarım sonucu (%)
Minkowski	3	91,9102
	5	95,7143
	7	95,2017
Öklid	3	91,9090
	5	95,7143
	7	95,0387
Manhattan	3	95,1709
	5	96,6667
	7	**97,1265**
Chebyschev	3	82,9013
	5	87,1429
	7	86,0722

Gerçekleştirilen çalışmada farklı komşuluk sayıları ve farklı mesafe ölçüm fonksiyonları kullanılmış olup, elde edilen başarım sonuçları Tablo 3'de verilmiştir. Elde edilen kümeleme sonuçları, veri setindeki sınıf bilgileri ile de kıyaslanmıştır.

K-en yakın komşuluk algoritmasında, yeni gelen verilerin komşuları ile arasındaki mesafeler ölçüldüğünde, seçilen k değeri çift sayı ise, hastalığa sahip kişilere ait veriler ile sağlıklı kişilere ait veriler arasındaki mesafe eşit çıkabilmektedir. Bu durumda kümeleme açısından problem oluşturabilmektedir. Bu nedenle çalışmada k değeri tek sayı seçilmiştir.

3.2. Destek vektör makineleri (Support vector machine)

Örnek sayısının fazla olması ve her bir örneğe ait 16 adet özniteliğin bulunması, veri kümesinin büyümesine neden olmuştur. Bunun sonucunda da SVM'nin eğitilmesinde ciddi hesaplama maliyeti oluşmaktadır. Bu hesaplama maliyetini en aza indirmek için sınıflandırma işlemi öncesinde öznitelik seçme işlemi gerçekleştirilmiştir. Öznitelik seçme işlemi sonucunda SVM'de kullanılan parametreler Tablo 4'de sunulmuştur. Öznitelik seçme işlemi uygulandıktan sonra elde edilen başarım sonuçları Tablo 5'de verilmiştir.

Tablo 4. SVM'de kullanılan parametreler (Parameters used in SVM)

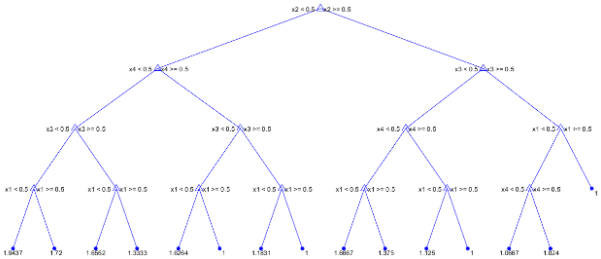
Parametre	Değeri
Çekirdek Fonksiyonu	Gaussian
Çekirdek Ölçeği	47,8300
Çözüm Fonksiyonu	Sıralı min. opt. (SMO)
İterasyon	1.000.000

Tablo 5. SVM yönteminden elde edilen başarım sonucu (Performance result from SVM method)

Giriş no	Başarım sonucu (%)
3	68,5714
11	67,6190
7,15	81,4286
8,9	85,7143
5,6,7	83,8095
9,10,11	82,8571
3,5,6,7	84,2857
13,14,15,16	68,5714
5,6,7,8, 9,10,11,12	91,4286
Tümü	**100**

3.3. Karar ağaçları (Decision trees)

Uygulamada çoklu sınıflandırma için CART (Classification And Regression Trees), ikili karar ağacı algoritması kullanılmıştır. Bu algoritmaya veri setindeki girişler, tek-tek ve farklı kombinasyonlar halinde girilmiştir. Elde edilen karar ağacı yapısında herhangi bir budama işlemi yapılmamıştır. Çalışmada iyi sonuç elde edilebilen örnek bir karar ağacı yapısı Şekil 2'de gösterilmiştir.



Şekil 2. Karar ağacı yapısı (Decision tree structure)

Bu yapıda maksimum karar sayısı 2, 3, 4 olarak test edilmiş olup, karar sayısı 2 olarak belirlendiğinde en iyi sonuç elde edilmiştir. Veriler için çapraz doğrulama işleminde de 10 değeri kullanılmıştır.

Elde edilen başarım sonuçları, veri setinin kullanılan girişleri, yaprak sayısı ve karar ağacının boyutu Tablo 6'da sunulmuştur.

Tablo 6. Karar ağacı sonuçları (Results of decision tree)

Giriş no	Yaprak sayısı	Ağacın boyutu	Başarım sonucu (%)
3	2	1	82,1703
11	25	13	78,6788
7,15	4	2	85,3668
5,6,7	8	3	91,0956
9,10,11	63	12	85,4653
3,5,6,7	15	4	91,9528
13,14,15,16	50	8	79,0225
5,6,7,8, 9,10,11,12	73	13	91,5030
Tümü	**49**	**9**	**93,3956**

Çalışmada gözetimli makine öğrenmesi modelleri uygulandıktan sonra makalede anlatılan sıra ile gözetimsiz öğrenme modelleri test edilmiştir.

3.4. K-Ortalamalar algoritması (K-Means algorithm)

Çalışmada kullanılan girişler sırasıyla tek-tek ve farklı gruplar halinde, K-Ortalamalar algoritmasında uygulanmıştır. Algoritma ilk çalıştırıldığında küme merkezleri rastgele atanmaktadır. Bu nedenle algoritma on kez arka arkaya çalıştırılarak sistemin ürettiği kümeleme başarım değerlerinin ortalaması alınmıştır.

Ayrıca kümeleme sonucunda elde edilen sınıf bilgileri, uzman kişilerce oluşturulan veri setindeki bilgiler ile kıyaslanmış olup, sonuçlar Tablo 7'de listelenmiştir.

K-Ortalamalar algoritmasında literatürde kullanılan farklı uzaklık fonksiyonları ile verdiği başarım sonuçları da Tablo 7'de sunulmuştur. Tablo 7'den de anlaşılacağı üzere bazı özelliklerin kümeleme performansını olumsuz yönde etkilediği görülmektedir.

Tablo 7. K-Ortalamalar algoritmasının sonuçları (Results of K-Means algorithm)

Giriş no	Uzaklık fonksiyonu	Başarım sonucu (%)
3	Öklid	57,5278
	Manhattan	36,0015
11	Öklid	53,5732
	Manhattan	51,2702
7,15	Öklid	79,1042
	Manhattan	17,3553
5,6,7	Öklid	86,6224
	Manhattan	12,4563
9,10,11	Öklid	56,0152
	Manhattan	49,3581
3,5,6,7	Öklid	81,8721
	Manhattan	**88,8069**
13,14,15,16	Öklid	50,7816
	Manhattan	46,7540
5,6,7,8, 9,10,11,12	Öklid	55,0871
	Manhattan	48,0047
Tümü	Öklid	54,1976
	Manhattan	45,8319

3.5. Temel bileşen analizi (Principal component analysis)

Temel bileşen analizi, veri setindeki anlamlı bilgileri ortaya çıkarmak için kullanılan, istatistiksel bir yöntemdir. Bu yöntemin genel amacı, verinin çeşitliliğini daha iyi yakalayacak yeni bir boyut takımının bulunmasıdır (Berkhin, 2002). Çalışmada kullanılan girişler ve sonuçları Tablo 8'de gösterilmiştir.

Tablo 8. PCA analiz sonuçları (Results of PCA analysis)

Giriş no	Doğru	Yanlış	Başarım sonucu (%)
3	728	326	69,0702
11	523	531	49,6205
7,15	764	290	72,4858
8,9	740	314	70,2087
5,6,7	797	257	75,6167
9,10,11	826	228	78,3681
3,5,6,7	864	190	81,9734
13,14,15,16	883	171	83,7761
5,6,7,8, 9,10,11,12	886	168	84,0607
Tümü	**889**	**165**	**84,3454**

3.6. Birliktelik kuralları (Association rules)

Birliktelik kural analizinin yapılabilmesi için kullanıcının en az destek ve güven değerlerini belirlemesi gerekmektedir. Bu değerlerin yüksek belirlenmesi düşük sayıda kuralın ortaya çıkmasına neden olmaktadır. Aynı durum bu değerlerin çok düşük belirlenmesinde de çok fazla kuralın ortaya çıkmasına sebep olmaktadır. Bu nedenle gerçekleştirilen testler neticesinde en az destek değeri 0,50 ve en az güven değeri de 0,25 olarak test edildiğinde elde edilen kural sayısının ideal olduğu tespit edilmiştir.

Tablo 9. Birliktelik analizi (Association rules analysis)

Kural no	X	Y	Destek(%)	Güven(%)
1	3	1	81,81	37,50
2	3	2	100	45,43
3	3	1,2	81,81	37,50
4	3	2,12	54,54	25,00
5	3	12	54,54	25,00
6	1,2	8	81,81	100
7	1,3	2	81,81	100
8	2,3	1	81,81	81,81
9	2,3	12	54,54	54,54
10	2,12	3	54,54	100
11	3,12	2	54,54	100

En az destek ve güven seviyesinde hesaplanan her bir kural Tablo 9'da listelenmiştir. Tablo 9'da 1 numaralı kurala göre, bütün veri setinin %81,81'inde çocuğun bir şey istediğinde onu işaret ettiği ve kendi ismiyle hitap edildiğinde baktığı görülmektedir. Güven değeri ise, bu çocukların %37,50'sinin OSB olma durumunu ifade etmektedir.

4. Sonuçlar (Conclusions)

Gerçekleştirilen çalışmada 12-36 ay arasındaki çocuklarda OSB teşhisinin yapılabilmesi için, gözetimli ve gözetimsiz olmak üzere toplam altı farklı makine öğrenme algoritması test edilmiştir. Yapılan bu testler sonucunda;

- K-En yakın komşuluk algoritmasında Manhattan uzaklık fonksiyonu ve 8 komşuluk ile %97,1265 başarım oranı elde edilmiştir.
- Destek vektör makinelerinde, Gaussian çekirdek fonksiyonu ve 47,83 çekirdek ölçeği ile veri setinde ki tüm girişler kullanılarak %100 başarım oranı elde edilmiştir. Buna karşın bütün özniteliklerin giriş olarak kullanılması, hesaplama maliyetini arttırmıştır.
- Karar ağacı yapısında, herhangi bir budama işlemi yapılmadan ve veri setindeki tüm özniteliklerin kullanılması ile 49 yapraklı 9 basamaklı bir ağaç yapısı oluşturulmuştur. Bu yapıda en fazla %93,3956 başarı oranı elde edilmiştir.
- K-Ortalamalar algoritması ile Tablo 2'de gösterilen 3, 5, 6, ve 7 nolu öznitelikler ile Manhattan uzaklık fonksiyonu kullanılarak %88,8069 başarım oranı elde edilmiştir.
- Temel bileşen analizi algoritmasıyla Tablo 2'de gösterilen tüm öznitelikler kullanılarak 1054 veriden 889 tanesi doğru, 165 tanesi yanlış olarak sınıflandırılarak en fazla %84,3454 başarım oranı elde edilmiştir.
- Birliktelik kuralları algoritmasında en az destek değeri 0,50 ve en az güven değeri de 0,25 olarak kullanıldığında, 6 ve 7 numaralı kurallara göre %81,81 destek oranı ile %100 güven ile gerçekleştirmek mümkün iken, 10. ve 11. kurallarda %54,54 destek oranı ile %100 güvenli sonuç bulmak mümkündür.

Yapılan çalışmadan da görüldüğü üzere, K-En yakın komşuluk algoritmasında (%97,1265) ve destek vektör makinelerinde (%100) yüksek başarı oranları elde edilmiştir. Elde edilen en iyi sonuçlardan ikisinin de gözetimli makine öğrenmesi yöntemi olması, kullanılan veri seti için gözetimli makine öğrenme yöntemlerinin, gözetimsiz makine öğrenme yöntemlerine göre daha yüksek başarım oranı sağladığını ortaya çıkarmaktadır.

Çalışma sonucunda, 12-36 ay arasındaki çocuklarda OSB teşhisinin yüksek doğrulukta yapılabilmesinde, ailelere ve uzman sağlık personeline yardımcı olabilecek bir yazılım geliştirilmiştir.

Kaynaklar (References)

Agrawal, R., Ramakrishnan, S., 1994. "Fast Algorithms for Mining Association Rules." Pp. 487–99 in Proc. 20th int. conf. very large data bases, VLDB. Vol. 1215.

Alpaydin, E., 2020. "Introduction to Machine Learning" MIT press.

Berkhin, P.,2002. 2nd. "Survey of Clustering Data Mining Techniques", Accrue Software Inc.

Cho, S., Mark, L., Neville, R., Meredith, C., Robert, T. S., Julia, P. M., 2019. "Automatic Detection of Autism Spectrum Disorder in Children Using Acoustic and Text Features from Brief Natural Conversations." Proc Interspeech. Graz, Austria.

Cover, T. M., Hart, P., 1967. "Nearest Neighbor Pattern Classification." IEEE Transactions on Information Theory IT13(1):21–27.

Çürükoğlu, N., 2019a. "Automated Demand / Suggestion Systems." Pp. 762–66 in 2019 4th International Conference on Computer Science and Engineering (UBMK).

Çürükoğlu, N., 2019b. "Imbalanced Dataset Problem in Classification Algorithms." Pp. 1–5 in 2019 1st International Informatics and Software Engineering Conference (UBMYK).

Dawson, G., Guillermo, S., 2019. "Potential for Digital Behavioral Measurement Tools to Transform the Detection and Diagnosis of Autism Spectrum Disorder." JAMA Pediatrics, 173(4):305–6.

Dogan, N., Ozdamar, K., 2003. "Chaid Analizi ve Aile Planlaması Ile Bir Uygulama." T. Klin Tıp.

Edureka. 2020. "Understanding K-Means Clustering with Examples." Retrieved (https://www.edureka.co/blog/k-means-clustering/).

Eker, M.E., Oktaş, R., Kayhan, G., 2015. "Apriori Algoritması ve Türkiye'deki Örnek Uygulamaları." Ondokuz Mayıs Üniversitesi Fen Bilimleri Enstitüsü, Samsun.

Eslami, T., Mirjalili, V., Fong, A., Laird, A.R., Saeed, F., 2019. "ASD-DiagNet: A Hybrid Learning Approach for Detection of Autism Spectrum Disorder Using FMRI Data" Frontiers in Neuroinformatics 13:70.

Huang, J., Lu, J., Ling, C.X., 2003. "Comparing Naive Bayes, Decision Trees, and SVM with AUC and Accuracy." Pp. 553–56 in Third IEEE International Conference on Data Mining. IEEE.

Kantardzic, M., 2011. Data Mining: Concepts, Models, Methods, and Algorithms. John Wiley & Sons.

Kayaalp, K., Metlek, S., 2020. "Otistik Spektrum Bozukluğunun Derin Öğrenme Yöntemi Ile Tespiti." P. 117 in 1. Uluslararası Sağlık Bilimlerinde Multidisipliner Çalışmalar Kongresi. İstanbul, Türkiye.

Lu, W., Wang, W., Leung, A.Y.T., Lo, S.M., Yuen, R.K.K., Xu, Z., Fan, H., 2002. "Air Pollutant Parameter Forecasting Using Support Vector Machines." Pp. 630–35 in Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No. 02CH37290). Vol. 1. IEEE.

Metlek, S., 2018. "Yapay Sinir Ağı Ile Otistik Spektrum Bozukluğunun Tespiti." Pp. 717–18 in I. Uluslararası Sağlık Bilimleri ve Yaşam Kongresi. Burdur, Türkiye.

Sayad, S., 2015. "Support Vector Machine - Classification (SVM)." http://ww.saedsayad.com/support_vector_machine.htm.

Seidman, C., 2001. Data Mining with Microsoft SQL Server 2000: Technical Reference. Vol. 7. Microsoft Press.

Shahamiri, R., Thabtah, F., 2018. "AutismAI: Autism Screening Mobile Application Based on Machine Learning." Retrieved (www.asdtests.com).

Shen, J., Pei. Z.J., Lee, E.S., 2004. "Support Vector Regression in the Analysis of Soft-Pad Grinding of Wire-Sawn Silicon Wafers." Pp. 19–24 in International Conference on Cybernetics and Information Technologies, Systems and Applications/ The 10th International Conference on Information Systems Analysis and Synthesis.

Taburoğlu, S., 2019. "A Survey on Anomaly Detection and Diagnosis Problem in the Space System Operation." Zeki Sistemler Teori ve Uygulamaları Dergisi 2(1):13–17.

Tan, P.N., Steinbach, M., Kumar, V., 2005. "Introduction to Data Mining: Pearson Addison Wesley." Boston.

Tan, P.N., Steinbach, M., Kumar, V., 2016. Introduction to Data Mining. Pearson Education India.

Taşcı, E., Onan, A., 2016. "K-En Yakın Komşu Algoritması Parametrelerinin Sınıflandırma Performansı Üzerine Etkisinin Incelenmesi." Akademik Bilişim.

Thabtah, F., 2018. "No Title." Retrieved (http://fadifayez.com/publications/#datasets).

Thabtah, F., Peebles, D., 2019. "A New Machine Learning Model Based on Induction of Rules for Autism Detection." Health Informatics Journal 1460458218824711.

Ülgen, E.K., 2017. "Makine Öğrenimi Bölüm-1." Retrieved (https://medium.com/bilişim-hareketi/ makine-öğrenimi-bölüm-1-f601b7225565).

Yıldız, E., Sevim, Y., 2016. "Comparison of Linear Dimensionality Reduction Methods on Classification Methods." Pp. 161–64 in 2016 National Conference on Electrical, Electronics and Biomedical Engineering (ELECO).

Yumuş, D., 2019. "Sar İmgelerinde Gözetimsiz Sınıflandırma Yöntemleri Ile Arazi Örtüsü Sınıflandırması."