Meme Kanseri Tanısında Wisconsin Veri Seti ile Makine Öğrenmesi Uygulamaları

Araştırma Makalesi/Research Article

Refik TANGİ[1] , Ramazan SOLMAZ[2]

[1]Bartın Üniversitesi, Lisansüstü Eğitim Enstitüsü, Akıllı Sistemler Mühendisliği Bölümü, Bartın, Türkiye
[2]Kahramanmaraş İstiklal Üniversitesi, Mühendislik Mimarlık ve Tasarım Fakültesi, Yazılım Mühendisliği Bölümü, Kahramanmaraş, Türkiye

refiktangi@windowslive.com, ramazan.solmaz@istiklal.edu.tr
(Geliş/Received:14.08.2024; Kabul/Accepted:05.11.2024)
DOI: 10.17671/gazibtd.1533288

Özet— Meme kanseri giderek daha sık görülmekte ve endişe verici bir boyuta ulaştığı ifade edilmektedir. Hastalık teşhis edilmezse ölüm riskini önemli ölçüde artırmaktadır. Son aşamada teşhis edildiğinde, tedbir olarak uzuvların alınması gerekmektedir. Erken teşhis için başarılı bir yöntem öncü olabilir. Bu makalenin odak noktası, meme kanseri teşhisinde başarılı makine öğrenimi tekniklerinin otomatik tanı için değerlendirilmesidir. Ayrıca, orijinal Wisconsin meme kanseri veri setine ait belirli özelliklerin etkinliği kontrol edilerek daha az işlem yükü ile başarılı tahminler araştırılmaktadır. Bu amaçla veri setine çeşitli makine öğrenimi algoritmaları uygulanmış ve en iyi performans gösteren algoritmalar belirlenmiştir. Daha başarılı bir tahmin için veri setine ön işlem uygulanarak etkin özellikler tespit edilmiştir. İlk bulgulardan yola çıkarak bu çalışmada, NB, DVM, J48 ve k-NN sınıflandırma algoritmaları ile k-means ve hiyerarşik kümeleme algoritmaları kullanılmıştır. Algoritmaların hastalık tanısındaki performansları doğruluk, ROC değerleri ve karmaşıklık matrisi metrikleriyle analiz edilmiştir. Performans metrikleri, en iyi sonucun NB tekniği ile elde edildiğini göstermektedir. Analiz edilen modellerin metrikleri, verilerin değerlendirilmesinde kullanılan çekirdek fonksiyonlarının tanıda önemli rol oynadığını göstermektedir. Wisconsin veri setine uygulanan denetimli algoritmalar güvenilir sonuçlar vermiştir. Meme kanseri teşhisinde başarılı olan algoritmaların sağlık sisteminde kullanılan analiz cihazlarına bir yazılım aracı olarak entegre edilmeleri, erken tanı ve farkındalık için iyi bir öncü olabileceği değerlendirilmektedir.

Anahtar Kelimeler— meme kanserinde otomatik tanı, öncü tasarımı, makine öğrenimi teknikleri, wisconsin veri seti

Machine Learning Applications on Wisconsin Dataset for Breast Cancer Diagnosis

Abstract— Breast cancer is increasingly common and is reaching an alarming level. If the disease is not diagnosed, it significantly increases the risk of death. When diagnosed at a late stage, the only precaution is often the removal of limbs. An effective method for early diagnosis could be a successful precursor. This paper focuses on evaluating successful machine learning techniques for automatic diagnosis in breast cancer detection. Additionally, the effectiveness of certain features of the original Wisconsin breast cancer dataset is examined to achieve accurate predictions with less computational load. For this purpose, various machine learning algorithms were applied to the dataset, and the best-performing algorithms were identified. To achieve more accurate predictions, preprocessing was applied to the dataset to identify effective features. Based on initial findings, NB, SVM, J48, and k-NN classification algorithms, as well as k-means and hierarchical clustering algorithms, were used in this study. The performance of the algorithms in disease diagnosis was analyzed using metrics such as accuracy, ROC values, and confusion matrices. Performance metrics indicate that the best result was obtained with the NB technique. The metrics of the analyzed models show that the kernel functions used in data evaluation play a significant role in diagnosis. Supervised algorithms applied to the Wisconsin dataset provided reliable results. It is considered that integrating successful algorithms in breast cancer diagnosis as a software tool into analysis devices used in the healthcare system could be a good precursor for early diagnosis and awareness.

Keywords— automatic diagnosis in breast cancer, precursor design, machine learning techniques, wisconsin dataset.

1. GİRİŞ (INTRODUCTION)

Meme kanseri, kadınlarda en yaygın görülen kanser türüdür ve kanser kaynaklı ölümlerin %15'ini oluşturmaktadır. Bu oranlar, gelişmekte olan ülkelerde daha yüksek seviyelerde seyretmektedir. Önümüzdeki 20 yıl içinde, teşhis konulan vaka sayısında (insidans) %55, ölüm oranlarında (mortalite) ise %58 artış öngörülmektedir [1-3]. Sağlık Bakanlığı istatistiklerine göre, meme kanseri ülkemizde görülen kanser türleri arasında ilk sıradadır [4]. 2019 yılında yaklaşık 4300 kadının meme kanseri kaynaklı hayatını kaybettiği ifade edilmiştir. Yine 2017 yılında yapılan "Ölüm Nedenlerinin Dağılımı" adlı araştırmada, Türkiye'de görülen ölümlerin %1'inin; AB ülkelerinde görülen ölümlerin ise %2'sinin meme kanseri kaynaklı olduğu rapor edilmiştir [2]. Kanser teşhisi alan her dört kadından birinin meme kanseri olduğu ve meme kanserinin giderek daha küçük yaşlarda görüldüğü not edilmiştir [5].

Meme kanserinin görülme sıklığı ve bu kanserin önemli sağlık sorunlarına yol açması nedeniyle, doktorların yanı sıra otomatik tanı için araştırmacıların da yoğun ilgisini çekmektedir. Erken tanı, hastalığın ilerlemesini durdurmada ve hastaların yaşam sürelerini uzatmada kritik bir rol oynamaktadır. Bu kapsamda makine öğrenimi yöntemleri, tıbbi tanı ve tedavi süreçlerinde giderek daha fazla kullanılmaktadır. Bu teknikler hastalık veri setleri üzerinde modelleme yaparak, hastalıkların erken tanısı ve doğru sınıflandırılması için güçlü araçlar sunmaktadır. Otomatik tanı aracı veya öncü tasarımı son zamanlarda araştırmacılar tarafından oldukça yoğun ilgi görmektedir. Wisconsin Meme Kanseri Veri Seti (Wisconsin Diagnostic Breast Cancer (WDBC)-Original) otomatik tanı araçlarını geliştirmek için yaygın olarak kullanılmaktadır. Wisconsin veri seti, biyopsi sonuçlarından elde edilen hücresel özellikleri içermekte olup, iyi huylu (benign) ve kötü huylu (malign) tümörlerin ayrımında kullanılmaktadır. Bu veri setine UCI makine öğrenimi veri tabanından ulaşılabilir [6].

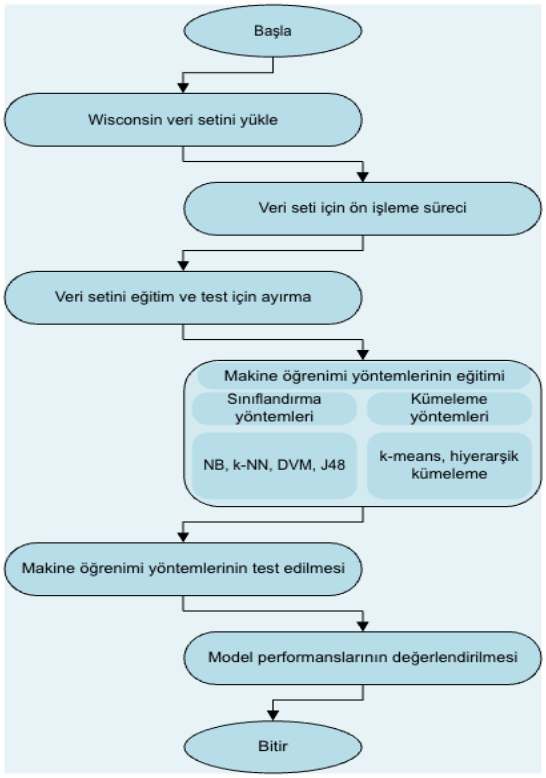
Makine öğrenimi algoritmalarının meme kanseri teşhisindeki etkinliği giderek artmakta ve algoritma performansını geliştirme çalışmaları devam etmektedir. Wisconsin veri setleri de meme kanseri teşhisinde sıkça başvurulan kaynak niteliğini taşımaktadır. Amrane ve arkadaşları (2018), bu veri setini kullanarak çeşitli denetimli öğrenme sınıflandırıcılarının performanslarını karşılaştırmış ve k-NN algoritmasının %97,51 doğruluk oranına sahip olduğunu belirtmişlerdir [7]. Benzer bir karşılaştırma yapan Aruna ve arkadaşları (2011), WDBC veri setine Destek Vektör Makinesi (DVM-RBF Kernel) sınıflandırıcı uygulayarak otomatik tanı konusunda %98,06'lık bir başarı derecesi elde ettiklerini ifade etmişlerdir [8]. Uddin vd., (2023) Wisconsin veri setinden yararlanarak özellik optimizasyonu tekniği ile makine öğrenimine dayalı meme kanseri teşhisi yapmışlardır. Sınıflandırıcıların performanslarının metrikler ile değerlendirildiği çalışmada en yüksek başarımı %98,77 ile oylama sınıflandırıcı (Voting classifier) ile elde ettiklerini belirtmişlerdir [9]. Nemade ve Fegade (2023) meme kanserinin kadınlarda ölümlerin ana nedenlerinden biri olduğunu ve meme kanseri tanısının oldukça zor olduğunu not etmişlerdir. Uzmanlar, kötü huylu tümör ile iyi huylu tümör arasında ayrım yapabilmek için bazı otomatik araçlara ihtiyaç duyulduğunu ifade etmişlerdir. Bu amaçla makine öğrenimi algoritmalarıyla Wisconsin veri seti örneklerini otomatik tasnif ederek tasnif işleminde kullanılan metotların performanslarını değerlendirmişlerdir. Çalışmalarında karar ağaçları ve XGBoost sınıflandırıcılar ile %97 başarı oranı elde ettiklerini belirtmişlerdir [10]. Singh vd. (2024) özellik seçiminin makine öğrenmesi algoritmaları performanslarında oldukça önemli olduğunu vurgulamış ve bunun için yeni bir algoritma (Feature selection-FS) önermişlerdir. Bu algoritma özellik seçimi için var olan iki algoritmayı birleştirmekte ve WDBC verilerini sınıflandırma için altı makine öğrenmesi metotlarını kullanmaktadır. Önerilen metodun uygunluğunu performans ölçüm metrikleri ile ölçerek elde edilen sonuçları literatür ile karşılaştırmışlardır [11]. Laghmati vd. (2023) makine öğrenme ve temel bileşen analizi (PCA) ile meme kanseri tahmin sistemlerini geliştirmeyi amaçlamakla birlikte kanser teşhisinde daha doğru ve hızlı sonuçlar elde etmeyi hedeflemişlerdir. Çalışmalarında k-NN algoritmasının %93,8 doğruluk oranına sahip olduğunu belirtmişlerdir [12]. Amethiya vd. (2021) makine öğrenimi yaklaşımına dayalı çeşitli algoritmaların ve biyosensörlerin erken meme kanseri tespiti için uygulanmasını araştırmak amacıyla çeşitli yaklaşımlar sunmaktadır ve bu kapsamda k-NN algoritmasının %95,9 doğruluk oranına sahip olduğunu ortaya koymaktadırlar [13]. Kadhim ve Kamil (2022) çeşitli kriterleri kullanarak makine öğrenimi algoritmalarının meme kanseri teşhisindeki başarıları araştırmışlardır. Bu amaçla çeşitli sınıflandırıcılar karşılaştırılmış ve ERT(Extremely randomized trees) algoritmasının %97,36 doğruluk oranına sahip olduğunu belirtilmişledir [14].

Bu çalışmada, meme kanseri veri setine etkinliği iyi bilinen Naive Bayes (NB), k-En Yakın Komşu (k-Nearest Neighbors, k-NN), J48 (Decision Tree), Destek Vektör Makinesi (DVM) gibi sınıflandırma yöntemleri ile hiyerarşik kümeleme ve k-means makine öğrenmesi algoritmaları uygulanarak otomatik tanı için yöntemlerin performansları ve doğruluk dereceleri ele alınmıştır. Meme kanseri veri seti 699 örnek, 10 öznitelik ve bir sınıf özniteliği içermektedir. Sınıf özniteliği, örneklerin kötü huylu ya da iyi huylu tümör bilgilerini içermektedir.

Bu çalışmanın odak noktası, veri seti analiz edilerek hastalık tespitinde etkin özellikleri belirlenmesidir. Ayrıca sürekli artan verilerden anlamlı bilgi çıkarmak için veri madenciliği süreci detaylı olarak ele alınmaktadır. Bu kapsamda algoritmaların sonuçları, geleneksel başarı oranı metrikleri üzerinden değil, daha detaylı ve yeni bir görüş olan her bir verinin kendi sınıfına ait olma durumu üzerinden de ele alınmaktadır. Algoritmaların performansları da daha detaylı analizi irdelenmiştir. Benzer şekilde literatürde bu veri seti için modellerin başarı oranı üzerinde durulurken, bu çalışmada başarılı bir tanı için önemli metrikler analiz edilmiştir.

2. MATERYAL VE METOT (MATERIAL AND METHOD)

Çalışmanın genel akış diyagramı Şekil 1'de verilmiştir. İlk olarak veri setine ön işlemler uygulanarak eğitim ve test kümesi olmak üzere iki gruba ayrılmaktadır. Daha sonra makine öğrenimi metotları eğitim kümesi ile eğitilmekte ve test veri seti ile test edilmektedir. Son olarak modellerin performansları metrikler ile değerlendirilmektedir.



Şekil 1. Metodoloji şematik diyagramı (Schematic diagram of methodology)

2.1. Materyal (Material)

Meme kanseri veri seti Kaliforniya Üniversitesi makine öğrenmesi veri tabanından alınmıştır [6]. Farklı zamanlarda elde edilmiş olan veri seti sekiz gruptan oluşmakta, toplam 699 örnekten(458 adet iyi huylu ve 241 adet kötü huylu) oluşan bu veri setinin gruplara göre dağılımı Tablo 1'deki gibidir.

Tablo 1. Veri Setinin Gruplara Dağılımı
(Distribution of the Data Set into Groups)

Grup	Örnek Sayısı	Yayın Tarihi
Grup 1	367	Ocak 1989
Grup 2	70	Ekim 1989
Grup 3	31	Şubat 1990
Grup 4	17	Nisan 1990
Grup 5	48	Ağustos 1990
Grup 6	49	Ocak 1991
Grup 7	31	Haziran 1991
Grup 8	86	Kasım 1991

Veri setinde bulunan her bir örnek 11 özellik içermekte ve örneklerde eksik bilgi bulunmamaktadır. Özellikler hücre boyutu ve şeklinin homojenlikleri, kanser hücresinin kümelenme derecesi, hücrelerin birbirine yapışması, epitel hücrelerinin boyutu, çekirdek yoğunluğu örnek kodu ve bölünme özellikleri iken son özellik bir sınıf bilgisidir. Sınıf bilgisi 2 ya da 4 seçeneklerinden oluşmaktadır. 2 seçeneği tümörün iyi huylu, 4 seçeneği tümörün kötü huylu olduğunu göstermektedir. Makine öğrenimi uygulamalarında tanı için kullanılan metotların başarımı etkileyen parametrelerin kolay ayarlanabilmesi, bu parametrelerin etkisinin kolay analiz edilebilmesi gibi avantajlar sunan Weka (3.8.6) yazılım aracı kullanılmıştır.

2.2 Metot (Method)

WDBC veri setine makine öğrenimi uygulamalarından doğruluk oranı yüksek olan k-NN, NB, J48 ve DVM gibi sınıflandırma algoritmaları ile kümeleme algoritmalarından k-means ve hiyerarşik kümeleme yöntemleri uygulanarak elde edilen sonuçlar ile algoritmaların performansları değerlendirilmiştir.

2.2.1 Sınıflandırma Yöntemleri (Classification Methods)

Sınıflandırma probleminin çözümü, otomatik sınıflandırma yapmak amacıyla nesnelerden oluşan veri kümesini test kümesi ve öğrenme kümesi olarak iki gruba ayırmaktadır. Burada nesneler niteliklerden oluşmakta ve niteliklerden biri ait olduğu sınıf bilgisini taşımaktadır. Sınıfın niteliğini belirlemek için tüm özellikler kullanılarak bir model oluşturulmaktadır. Ardından, test kümesinde bulunan ve sınıfı bilinmeyen nesneler, oluşturulan model kullanılarak en uygun sınıflara atanmaktadır. Yani, bağımsız değişkenler için sınıf tahmini yapılmaktadır [15].

Çalışmada kullanılacak sınıflandırıcıların seçimi için yapılan ön uygulamalar kapsamında başarı oranı yüksek olan k-NN, NB, J48 ve DVM algoritmaları seçilmiştir. Tüm sınıflandırıcılar için, en çok 10 kat çapraz doğrulama (cross validation) verimli olmuştur. Çapraz doğrulama, genellikle öğrenme algoritmalarını veya modellerini kontrol etmek ve değerlendirmek için kullanılan istatistiksel bir tekniktir. Bu teknik, veriyi belirlenen sayıda gruplara ayırmaktadır, bu veri grubundan bir tanesini modeli test etmek ve diğerlerini modeli eğitmek için kullanmaktadır.

Sınıflandırıcı performans değerlendirmesi için hata matrisi ve bu matrisle hesaplanan performans ölçüm teknikleri olan F1 skorlama, doğruluk, duyarlılık, kesinlik metrikleri kullanılmaktadır. Bu çalışmada performans değerlendirmesinde kullanılan teknikler ve formüller aşağıda verilmektedir [16].

Karmaşıklık matrisi (Confusion matrix): "Hata matrisi" veya "karmaşıklık matrisi", bir sınıflandırma modelinin performansını ölçmek için yaygın kullanılan araçlardandır. Bu matris verilerin gerçek sınıf etiketleri ile modelin tahmin ettiği sınıf etiketlerinin birbirleriyle

karşılaştırılmasını sağlar. Karşılaştırma sonucunda, doğru sınıflandırılan örnekler (doğru pozitifler ve doğru negatifler) ile yanlış sınıflandırılan örnekler (yanlış pozitifler ve yanlış negatifler) arasındaki ilişki net bir şekilde gösterilmektedir. Karmaşıklık matrisi, sınıflandırma problemlerinde modellerin performans değerlendirmesinde tek başına kullanılan bir ölçüt olmamakla birlikte genellikle şu dört temel değeri içerir.

Doğru Pozitif (DP): Modelin doğru bir şekilde pozitif olarak sınıflandırdığı örneklerin sayısı.
Yanlış Pozitif (YP): Modelin yanlış bir şekilde pozitif olarak sınıflandırdığı örneklerin sayısı.
Doğru Negatif (DN): Modelin doğru bir şekilde negatif olarak sınıflandırdığı örneklerin sayısı.
Yanlış Negatif (YN): Modelin yanlış bir şekilde negatif olarak sınıflandırdığı örneklerin sayısı.

Bu değerler, modelin doğruluğunu, hassasiyetini, özgüllüğünü ve duyarlılığını hesaplamak için kullanılmaktadır. Karmaşıklık matrisi, modelin performansını anlamak ve geliştirmek için önemli bir araçtır.

Doğruluk (Accuracy): Doğruluk, modelin doğru tahminlerin toplam tahminlere oranıdır. Yüksek bir doğruluk, modelin genel olarak doğru tahminler yaptığını gösterir. Ancak, dengesiz sınıf dağılımına sahip veri kümelerinde doğruluk tek başına yeterli olmayabilir. Örneğin 1000 örnek içeren bir veri setinde a sınıfına ait 990 ve b sınıfına ait 10 örnek olsa yapılacak bir sınıflandırmada işleminde verilerinin tamamının a sınıfına atanması durumunda bu işlemde başarı oranı %99 olacaktır. Örnekte görüldüğü gibi bir modelin performansını değerlendirmede doğruluk kriteri tek başına yeterli görünmemektedir. Karmaşıklık matrisi sonuçları Denklem 1'de verilen ifade ile değerlendirilerek yöntemin doğruluk oranı elde edilir.

$$\text{Doğruluk} = \frac{\text{DP}+\text{DN}}{\text{DP}+\text{DN}+\text{YP}+\text{YN}} \quad (1)$$

Kesinlik (Precision): Kesinlik veya hassasiyet, pozitif olarak tahmin edilen örneklerin gerçekten pozitif olma oranını gösterir. Yüksek hassasiyet, yanlış pozitiflerin az olduğunu ve modelin yanlış alarm verme olasılığının düşük olduğunu gösterir. Özellikle yanlış pozitiflerin maliyeti yüksek olduğunda önemlidir. Denklem 2'de verilen eşitlik ve karmaşıklık matrisi kullanılarak modelin kesinlik oranı elde edilir.

$$\text{Kesinlik} = \frac{\text{DP}}{\text{DP}+\text{YP}} \quad (2)$$

Duyarlılık (Recall, Sensitivity): Pozitif olarak tahmin etmemiz gereken örneklerin ne kadarını pozitif olarak tahmin ettiğimizi gösteren bir metriktir. Özellikle Yanlış Negatif'e odaklanıldığında duyarlılık hesabı önem kazanmaktadır. Denklem 3 ve 4'te verilen ifadeler karmaşıklık matrisi verilerine uygulanarak duyarlılık ile Özgüllük oranı hesaplanmaktadır.

$$\text{Duyarlılık} = \frac{\text{DP}}{\text{DP}+\text{YN}} \quad (3)$$

$$\text{Özgüllük} = \frac{\text{DN}}{\text{DN}+\text{YP}} \quad (4)$$

F1 Skorlama (F1 score): Homojen dağılıma sahip olmayan veri setlerinde model performansı analizinde kullanılan F1 skorlama, hassasiyet ve duyarlılığın harmonik ortalaması olarak tanımlanır. Dengesiz veri setlerinde kesinlik ve duyarlılık performans ölçümlerinde kullanılan önemli ölçütlerdendir. F1 skorlama ise kesinlik ve duyarlılık ölçütlerini kullanarak dengesiz veri setleri için model performansını daha doğru bir şekilde ölçmektedir. Homojen dağılıma sahip olmayan veri setlerinde ve yanlış pozitifler ile yanlış negatifler arasında bir denge kurmak istendiğinde faydalıdır. Yüksek bir F1 skoru, hem yanlış pozitiflerin hem de yanlış negatiflerin az olduğunu ve dolayısıyla modelin genel olarak iyi performans gösterdiğini belirtir. Denklem 5'te verilen ifade karmaşıklık matrisi verilerine uygulanarak modelin F1 skorlama oranı tespit edilir.

$$\text{F1 Skorlama} = 2 * \frac{\text{Duyarlılık}*\text{Hassasiyet}}{\text{Duyarlılık}+\text{Hassasiyet}} \quad (5)$$

2.2.1.1 k-NN Sınıflandırıcı Algoritması (k-NN Classifier Algorithm)

k-NN, sınıflandırma problemi çözümü için kullanılan en sade makine öğrenmesi algoritması olarak kabul edilebilir. Temel prensibi, bir veri noktasını sınıflandırmak için çevresindeki k adet en yakın komşusunun sınıf bilgisine dayanır. k-NN, öğrenme süreci içinde veri setini öğrenmez; bunun yerine, sınıflandırma yapılması istenen yeni bir veri noktası geldiğinde, bu noktaya en yakın k adet komşularının verilerini kullanarak sınıflandırmayı gerçekleştirir. Sınıflandırma işlemi yapılırken komşulukların çoğunluğu dikkate alınmaktadır. Eşitlik olmaması adına k değeri genellikle pozitif tek sayı olarak belirlenir. Sınıflandırılmak istenen yeni verinin, mevcut verilere olan uzaklığı hesaplanıp, k sayıda en yakın komşuluğuna bakılmaktadır. Mesafe hesaplamaları için birçok mesafe fonksiyonu kullanılmaktadır. Literatürde yaygın olarak kullanılan mesafe ölçütleri Oklid, Minkowski ve Manhattan uzaklıklarıdır.

k-NN algoritması, parametrik olmayan bir tembel öğrenme (lazy learning) algoritmasıdır. Lazy learning'in bir eğitim aşaması yoktur; yani eğitim verilerini öğrenmez, bunun yerine eğitim veri kümesini ezberler. Bir sınıflandırma işlemi gerçekleştirmek istendiğinde, tüm veri seti içerisinde en yakın komşuları arar. Algoritmanın çalışmasında bir k parametresi belirlenir. Bu k parametresi bir veri noktasını sınıflandırmak için çevresindeki k en yakın komşusunu seçmek için kullanılır. Yeni bir değer geldiğinde en yakın k adet eleman alınarak gelen değer arasındaki uzaklık hesaplaması yapılır. Uzaklık hesaplama işlemlerinde genelde en yaygın olarak Öklid mesafe fonksiyonu kullanılmaktadır. Öklid fonksiyonu haricinde Manhattan ve Minkowski fonksiyonları da alternatif olarak kullanılabilir. Belli fonksiyonlara göre uzaklık hesabı

yapılmakta, ardından uzaklık hesapları sıralanmakta ve gelen değerler uygun olan sınıfa atanmaktadır [17,18]. Yaygın kullanılan uzaklık hesaplama formülleri Denklem 6, 7 ve 8'de verilmiştir. Bu formüller, ilgili veri setindeki her bir örnek sınıflandırılırken, yeni bir veri noktasının hangi sınıfa ait olduğunu, komşularıyla olan mesafeyi hesaplayarak belirlemektedir.

$$\text{Öklid (Euclidean)} = \sqrt{\sum_{i=1}^{k}(x_i - y_i)^2} \quad (6)$$

$$\text{Manhattan} = \sum_{i=1}^{k}|x_i - y_i| \quad (7)$$

$$\text{Minkowski} = \left(\sum_{i=1}^{k}(|x_i - y_i|)^p\right)^{1/p} \quad (8)$$

2.2.1.2 Naive Bayes Sınıflandırıcı Algoritması (Naive Bayes Classifier Algorithm)

Naive Bayes (NB) sınıflandırıcısının temeli Bayes teoremine dayanan istatistiksel sınıflandırma problemlerinde kullanılan bir makine öğrenimi algoritmasıdır. Tembel öğrenme algoritmalarından biri olan NB dengesiz veri kümelerinde de çalışabilmektedir. Algoritma, sistemdeki değişikliklere kendini adapte edebilir başka bir ifade ile yeni gelen örnekler olduğunda değişikliklere duyarlı olabilmektedir. Algoritmanın çalışma şekli, ilgili örnek için her bir durum olasılığı hesaplanarak olasılık değeri en yüksek olan sınıfa göre sınıf bilgisi belirlenmektedir. Örneklerin hangi sınıfa ve hangi olasılıkla ait olduklarını belirleyen NB, düşük boyutlu veri setiyle başarılı sonuçlar ortaya koyabilmektedir. Eğitim kümesinde belirlenemeyen bir değer varsa, model test kümesinde tahmin yaparken bu değer için genellikle bir olasılık değeri verilmez (değeri 0 olarak verir) ve tahmin yapılamaz çünkü olasılık hesabında sonucun 0 çıkmasına (ilgili veride 0 kayıt olması durumuna) sebep olmaktadır. Bu durum, Sıfır Frekans yani Zero Frequency adıyla da bilinir. Bu sorunu çözebilmek için çeşitli düzeltme yöntemleri kullanılmakla birlikte Laplace yöntemi (Denklem 9) en basit düzeltme yöntemlerinden biri olarak öne çıkmaktadır [19,20]. Laplace formülü ile her bir özellik için sınıfa ait olasılıklar çarpılır ve sonuçlar karşılaştırılarak en yüksek olasılığa sahip sınıf seçilmektedir.

$$\text{Laplace Formülü: } P(X|C_i) = \prod_{k=1}^{n} P(X_k|C_i) \quad (9)$$

NB sınıflandırıcısının davranışını değiştirmek ve sınıflandırma performansını artırmak/iyileştirmek için useKernelEstimator ve useSupervisedDiscretization parametreleri optimize edilir. "useKernelEstimator", normal dağılım yerine sayısal nitelikler için çekirdek kestirimcisi kullanan bir parametredir. Weka'da varsayılan olarak her sayısal özellik için Gauss dağılımı kabul edilir. Algoritma, useKernelEstimator argümanı ile çekirdek kestirimcisini kullanacak şekilde değiştirilebilir, bu da veri kümesindeki niteliklerin gerçek dağılımıyla daha iyi sonuçlar verebilir. "useKernelEstimator" parametresi, bir boolean (true/false) değer alır. Varsayılan olarak, bu parametre pasif (false) seçeneğindedir, yani çekirdek tahmincisi kullanılmaz. useSupervisedDiscretization parametresi ise değişken dönüştürme işleminin kullanılıp kullanılmayacağını belirler yani sayısal özellikleri nominal özelliklere otomatik olarak dönüştürebilmektedir. Bu parametrenin varsayılan değeri pasif olup, veri setindeki sürekli özelliklerin sürekli değerlerini belirli aralıklara bölerek verileri sınıflandırmaya yardımcı olur. Söz konusu parametrelerin farklı değerlerle deneysel olarak test edilmesi ve performanslarının değerlendirilmesi önemlidir. Çünkü bazen bu parametreler veri seti üzerinde olumsuz etki oluşturabilmektedir [21].

2.2.1.3 J48 (Karar Ağacı) Sınıflandırıcı Algoritması (J48 (Decision Tree) Classifier Algorithm)

Bu algoritma, karar ağacını daha sade, daha küçük, daha iyi optimize edilmiş ve verimli hale getirmeyi amaçlamaktadır. Bunun için de değişkenlerin/özelliklerin entropi ve bilgi kazanımı (information gain) değerlerini esas alır. Entropi ve bilgi kazanımı konuları literatürde geniş bir şekilde bulunduğu için aşağıda konu hakkında özet bilgilere verilmiştir.

Entropi, rastgele bir değişkenin belirsizliğinin ölçüsü olarak ifade edilmektedir.

Bilgi Kazanımı: Veriler bölümlendiğinde, hedef değişkendeki belirsizliğin ne kadar değiştiğinin ölçüsüdür. Başka bir ifade ile yeni bilgilerin öğrenilmesi olarak ifade edilebilir.

C4.5 veya J48 algoritması, ilk olarak hedef değişken için entropi değerini hesaplar. Daha sonra, her bir değişkenin veya özelliğin bilgi kazanımını hesaplar ve bu şekilde en yüksek bilgi kazanımı değerine sahip tahmin edici sınıfı tespit eder [22]. En yüksek bilgi kazanımı değerine sahip özellik/değişken, ağacın en üst nodunda (düğüm) yer almaktadır. Yani en iyi bilgi kazanımı sonucunu veren özellik/değişken, karar(dallanmanın başladığı nokta) olarak belirlenir. Ardından alt düğümler için de tüm özelliklere/değişkenlere aynı işlemler tekrarlanır [23]. Bu şekilde karar ağacının daha dengeli bölünmesi beklenmektedir. Bu hesaplamalarda kullanılan formüller ise Denklem 10, 11 ve 12'de verilmiştir. Denklem 10'da verilen eşitlik yardımıyla veri kümesindeki belirsizlik veya düzensizlik (entropi) belirlenmektedir. Değişkenin bilgisi (Denklem 11), bir değişkenin (özelliğin) veri kümesindeki belirsizliği ne kadar azalttığını ifade ederken bilgi kazanımı (Denklem 12) ise bir değişkenin sınıflandırmada ne kadar fayda sağladığını göstermektedir. Başarılı bir sınıflandırma, veri bir özelliğe göre bölündüğünde entropinin minimum ve bilgi kazanımının maksimum olması beklenir.

$$\text{Entropi : } \text{Info}(D) = -\sum_{i=1}^{m}(p_i \log_2 p_i) \quad (10)$$

$$\text{Değişkenin Bilgisi: } \text{Info}_A(D) = -\sum_{j=1}^{V}\left(\frac{|D_j|}{D} \text{x Info}(D_j)\right) \quad (11)$$

$$\text{Bilgi Kazanımı: } \text{Gain}(A) = \text{Info}(D) - \text{Info}_A(D) \quad (12)$$

Sistemin aşırı eğitilmesi (overfitting) sonucunda esnekliğini kaybetmesi ve ezberlemeye yakın bir sonuç oluşturması, gereksiz yere fazla detay içermesi muhtemeldir. Bu durumu önlemek için ağaç dallarında budama (pruning) yapılır. Ağaç oluşturulurken ön budama (prepruning) ve ağaç oluşturulduktan sonra ise son budama (postpruning) yapılmaktadır. Bu sayede aşırı öğrenmenin önüne geçilebilmektedir.

J48 algoritmasının aşağıda bulunan bazı parametreleri sınıflandırmadaki başarı oranını etkileyebilmektedir:

Güven Faktörü (confidenceFactor): Bu parametre, karar ağacının dengeli bir şekilde büyümesine katkıda bulunan budama işleminin etkinliğini artırmak için kullanılır. Bu parametre küçük değerler aldığında, daha fazla budama işlemi gerçekleştirilmesine olanak tanır [24].

MinNumObj: Her yaprakta/düğümde bulunması gereken kayıt sayısı verisini belirtmektedir.

doNotMakeSplitPointActualValue: Algoritmanın hangi değerleri ayrım noktası (split point) olarak kullanılmaması gerektiğini belirlemesini sağlar. Karar ağaçları, veri kümesindeki belirli özelliklerin (features) değerlerine göre veri noktalarını bölümlere ayırır. Bu bölme işlemi, belirli bir eşik değer (split point) kullanılarak gerçekleştirilir. Ancak bazen bu eşik değerlerinin belirli bir aralıkta veya belirli bir değerin altında veya üstünde olması istenmez. "*Dont make split point actual value*" parametresi, bu tür durumlarda kullanılır. Bu parametre, algoritmanın belirli bir eşik değeri kullanarak bölme yapmasını engellemek için kullanılır. Örneğin, belirli bir özelliğin değeri 0 ile 100 arasında değişiyorsa, ve bu parametre 50 olarak ayarlanırsa, algoritma bu özellik için 50'nin bir split point olarak kullanılmasını engeller. Bu parametre, belirli bir özellik için kullanılabilecek tüm değerlerin ayrım noktası olarak kullanılmasını önleyerek modelin genelleştirilebilirliğini artırabilir veya istenmeyen ayrışmaları engelleyebilir. Bu nedenle, veri kümesine ve problem alanına bağlı olarak, bu parametrenin kullanımı modelin performansını artırabilir veya istenmeyen sonuçları önleyebilir.

2.2.1.4 Destek Vektör Makinesi Sınıflandırıcı Algoritması (Support Vector Machine Classifier Algorithm)

DVM sınıflandırma ve regresyon problemleri için kullanılan bir makine öğrenimi algoritmasıyken aynı zamanda bir optimizasyon algoritmasıdır. Bir makine öğrenimi algoritması olan DVM genellikle çift özdeş olmayan doğrusal programlama problemlerini çözmek için kullanılan optimizasyon tekniklerini içerir. Aynı zamanda özel bir optimizasyon algoritmasıdır, büyük problemleri daha küçük alt problemlere bölerek çözme özelliğine sahiptir. Bu algoritma, özellikle büyük veri setleri veya yüksek boyutlu özellik uzayları gibi durumlar için etkili bir seçenek olabilir.

DVM'nin çalışma prensibi, sınıflandırma için bir düzlemde bulunan veri kümeleri arasına sınırlar çizilerek gruplara ayırmak mümkündür. Bu sınırlar gruplar arasındaki en uzak yere çizilmektedir [25]. DVM algoritması her bir sınıfın en yakın veri noktaları arasında en fazla mesafeye sahip olan hiperdüzlemi bularak başlar. Hiperdüzleme en yakın mesafede olan veri noktaları ise destek vektörleri olarak tanımlanır ve hiperdüzlemi tanımlamak için kullanılır. Algoritma bu destek vektörlerini kullanarak veriler arasındaki farklı sınıfları ayıran bir karar sınırı oluşturur. Belki de iki sınıfı ayıracak sonsuz sayıda doğru çizilebilir fakat DVM iki grup/sınıf arasında en fazla aralığa sahip olan doğruyu seçer [26,27].

DVM algoritması, yeni bir veri noktasını sınıflandırmak için hiperdüzlemdeki konumunu (hiperdüzlemin hangi tarafına düştüğünü) kullanır. Veri noktasının hiperdüzlemin hangi tarafında bulunduğuna bağlı olarak, ilgili sınıfa atanır yani hiperdüzlemin bir tarafına düşer ise bir sınıfa ait olarak sınıflandırılır, diğer tarafa düşer ise diğer sınıfa ait olarak sınıflandırılır. Bu, DVM'lerin, regresyon ve sınıflandırma gibi denetimli öğrenme görevlerinde güçlü bir araç olmasını sağlar. Özellikle, yüksek boyutlu veri alanlarında, farklı sınıfları ayıran veya gerçek ve tahmin edilen değerler arasındaki hatayı en aza indiren bir hiperdüzlem bulma stratejisi izlerler. Bu şekilde, verileri daha iyi anlamak ve öngörülemeyen desenleri keşfetmek için kullanılabilirler [28]. Düzlem ve boyutlar birer özellik (attribute) olarak düşünülebilir, her girdiyi gösteren farklı bir nokta elde edilmektedir ve ardından bu girdiler sınıflandırılmaktadır. Doğrusal (Linear) kernel haricinde polykernel, RBF (Radial Basis Function) gibi fonksiyonlar da kullanılabilmektedir.

Çok sınıflı örneklerde her bir sınıf arasında ayırma yapılmakta ve yeni gelen örnek buna göre sınıflandırılmaktadır. DVM'nin önemli parametresi olan "C" (Cost), sınıflandırıcı performansını etkileyen kritik bir faktördür.

C (Cost) Parametresi:
Amacı: C değeri, DVM'nin düzenleme gücünü kontrol eder. Yüksek C değerleri, eğitim verilerine daha fazla vurgu yapılmasını ve karar sınırlarının daha fazla düzenlenmesini sağlar, bu da modelin eğitim verilerine daha sıkı uymasına neden olabilir.

Etkisi: Yüksek *C* değerleri overfitting eğilimindedir, yani model eğitim verilerine çok fazla uyar ve genelleme yeteneği düşer. Düşük *C* değerleri, daha genel geçer modellere yol açabilir, ancak eğitim verilerine daha az uyarlanabilirler.

2.2.2 Kümeleme Yöntemleri (Clustering Methods)

Kümeleme Yöntemleri, keşifsel veri analizi yöntemi olup, bir dizi veri öğesini, bir uzaklık (veya benzerlik) ölçüsüne dayalı olarak gruplara/kümelere/bölütlere ayırmayı amaçlamaktadır. Ya da kısaca birbiriyle özdeş/yakın özellikte olan verilerin tek bir grupta toplanması olarak da

ifade edilebilmektedir. Bu gruplara "küme" denir ve sayıları önceden belirlenebileceği gibi algoritmalar tarafından da belirlenebilmektedir. Burada ilgili veri seti için kümeleme sonuçları incelenirken ayrıca Classes to Clusters Evaluation seçeneği ile değerlendirme yapılmaktadır. "Classes to Clusters Evaluation", kümeleme (clustering) işlemi sonucunda oluşturulan kümelemelerin sınıflandırma (classification) hedefiyle karşılaştırılmasını sağlayan bir değerlendirme yöntemidir. Bu işlem, genellikle denetimli öğrenme (supervised learning) ve denetimsiz öğrenme (unsupervised learning) yöntemlerini bir araya getirir.

Söz konusu yöntem, iki ana bileşenden oluşur:

Sınıflar (Classes): Veri kümesindeki örneklerin ait olduğu sınıfları belirtir. Sınıflar genellikle bir etiket veya kategori olarak temsil edilir ve her bir veri örneği için bir etiket atanır.

Kümelemeler (Clusters): Veri kümesindeki örneklerin benzerliklerine dayanarak oluşturulan veri gruplarını temsil eder. Kümeleme algoritmaları, veri örneklerini birbirine benzer olan gruplara böler.

"Sınıfların Kümelemelere Atanması (Classes to Clusters)" değerlendirmesi, bu iki bileşeni karşılaştırır ve sınıfların, kümelemelerle nasıl ilişkilendirildiğini değerlendirir. Genellikle, her sınıfın hangi kümelere ait olduğunu ve kümeleme sonuçlarının sınıflandırma doğruluğunu ölçer.

Bu işlem, doğru sınıfların doğru kümelere atanmasını göz önünde bulundurarak kümeleme algoritmasının performansını ölçer. İdeal olarak, her sınıfın bir kümeyle tam olarak eşleşmesi ve sınıflandırma doğruluğunun maksimum düzeyde olması beklenir. Ancak, gerçek dünyada bu durum genellikle mümkün olmayabilir ve bu nedenle değerlendirme işlemi, sınıfların ve kümelemelerin ne kadar iyi eşleştiğini nicel olarak ölçmeye çalışır.

Sınıflandırma bilgisine dayalı kümeleme problemlerinde algoritmalarının performansını anlamak ve geliştirmek için bu değerlendirme işlemi önemli olmaktadır.

2.2.2.1 k-Means Kümeleme Algoritması (k-Means Clustering Algorithm)

k-means algoritması, kümeleme algoritmalarının içinde muhtemelen en eski ve yaygın olarak kullanılan basit bir algoritmadır. Eğitimsiz/Eğiticisiz (Unsupervised) öğrenme prensibine sahiptir. Bu algoritmanın avantajları ve dezavantajları bulunmakla birlikte, büyük veri kümelerinde hızlı çalışması nedeniyle popülerlik kazanmıştır.

k-means algoritmasında, kümelenecek olan verilerden her biri yalnızca bir küme üyesi olarak atanabilir ki bu da disjoint kümeleme mantığını ifade etmektedir. Disjoint kümelemede kümeler birbirinden belli bir hatla ayrılabilen, birbirine girmemiş kümeleme yöntemidir. Bu kümelerin temsil edildiği noktalar ise merkez noktasını göstermektedir. Bu algoritmada kullanılacak verinin bölüneceği küme sayısını, kullanıcının manuel girmesine bağlı olarak belirlemesi durumu bulunmaktadır, “numCluster” parametresi bu işe yaramaktadır yani kullanıcının belirlemesi gereken ve k-means algoritmasının veri noktalarını kaç kümeye böleceğini belirten bir parametredir. Bu sebeple doğru küme sayısı belirlenme durumu bitene kadar deneme yanılma yöntemine başvurulması gerekebilmektedir. Bazen k-means işleminin başarılı şekilde tamamlanması için fonksiyonun birkaç kez çağırılması gerekebilmektedir. Çünkü kümelerin içinde ilk seferde oluşan benzerlik uyumu doğru sonuç vermeyebilir/tutmayabilir. Daha sonra kümelerde değişimin durması yani uyumun tutması istenilen sonucun alındığı anlamına gelebilmektedir [29,30].

Kısaca bu algoritmada, veriler iki boyutlu uzay üzerine serilir. Hedef kümeler tanımlanır, ardından belli mesafe fonksiyonlarına göre (Öklid, Manhattan vb.) örneklerin hedeflere mesafesi hesaplanır. Hedefler belirlenirken rastgele başlangıç merkezlerinin belirlenmesinde “seed” başlangıç değeri kullanılmaktadır, daha sonra merkezin etrafındaki örnekler kümeye dahil oldukça merkez nokta değişebilmektedir. Tüm kümeleme işlemi bu adımlarla tamamlanmaktadır [29].

2.2.2.2 Hiyerarşik Kümeleme Algoritması (Hierarchical Clustering Algorithm)

Hiyerarşik kümeleme, parametrelerin belirsiz olduğu durumlarda ya da kaç bölütün/kümenin oluşturulacağının bilinmediği veya duruma göre değiştiği problemlerde kümeleme yerine hiyerarşi oluşturmaya yardımcı olan algoritmadır.

Hiyerarşik algoritmalarda aşağıdan yukarıya (Agglomerative/AGNES) ve yukarıdan aşağıya (Divisive/DIANA) olmak üzere iki adet yöntem bulunmaktadır. Aşağıdan yukarıya kümeleme mantığına göre, verilerin her biri başlangıç aşamasında tekil bir küme olarak ele alınır ve benzerlikleri en yüksek olan veri noktaları bir araya getirilerek kümelenir. Kümeleme işlemi için örnekler/özellikler arasındaki ilişkilere göre ilk olarak ikili ilişkileri içeren bağlantılara göre ve ardından daha fazla sayıdaki ilişkileri içeren bağlantılara göre hiyerarşi oluşturulmaktadır. Bu işlem kümelenecek/bölütlenecek başka bir veri kalmayıncaya kadar devam etmektedir. Sonuç ağacı dendrogram ile gösterilmektedir [30,31]. Yukarıdan aşağı kümeleme yaklaşımda örneklerin tamamı bir bütün olarak ele alınmakta ve alt gruplara bölünmektedir. Örneğin, önce iki gruba bölünmekte sonra her grup kendi içinde daha alt gruplara bölünerek ilerlenmekte ve sonuç ağacı yine dendrogram ile gösterilmektedir [31].

2. BULGULAR (FINDINGS)

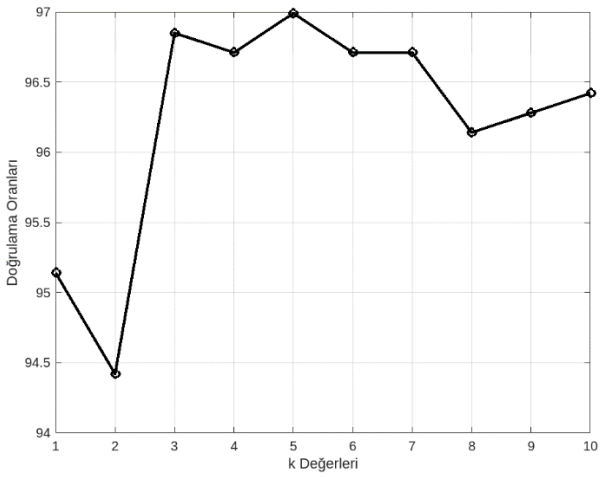
Hastalık verilerini sınıflandırma ve kümeleme için modeller, bağımsız uygulamalarda eğitim veri seti ile eğitilerek test veri seti ile performansları değerlendirilmektedir. Uygulamalardan elde edilen bulgular alt başlıklarda verilmektedir.

3.1 k-NN Sınıflandırıcı Uygulaması (k-NN Classifier Application)

k-NN sınıflandırıcısındaki k değeri (kullanılacak komşu sayısı) etkisinin incelendiği sonuçlar Tablo 2 ve Şekil 2'de sunulmuştur. Bu sonuçlara göre, k = 5 koşulu en yüksek doğruluğa sahiptir. Buna göre %96,99'luk bir doğrulama oranı mevcuttur.

Tablo 2. Farklı k değerleri için *k*-NN sınıflandırıcının sınıflandırma ve doğrulama sonuçları
(Classification and validation results of k-NN classifier for different k values)

k Değeri	Doğru Sınıflandırma	Yanlış Sınıflandırma	Doğruluk Oranı(%)
k=1	665	34	95,14
k=2	660	39	94,42
k=3	677	22	96,85
k=4	676	23	96,71
k=5	678	21	96,99
k=6	676	23	96,71
k=7	676	23	96,71
k=8	672	27	96,14
k=9	673	26	96,28
k=10	674	25	96,42



Şekil 2. *k*-NN sınıflandırıcının *k=1-10* değerleri için başarı oranları (Success rates of k-NN classifier for k=1-10 values)

k=5 değeri için en iyi doğrulama sonucunun alındığı k-NN sınıflandırıcıya ait karmaşıklık matrisi Tablo 3'te verilmiştir.

Tablo 3. k=5 değeri için karmaşıklık matrisi
(Complexity matrix for k=5)

k Değeri	a	b	
k=5	444	14	**a=2**
	7	234	**b=4**

* a veya 2 değeri iyi huylu tümörü, b değeri veya 4 kötü huylu tümörü temsil etmektedir

Tablo 4. *k*-NN sınıflandırıcı için performans değerlendirme ölçütleri
(Performance evaluation metrics for k-NN classifier)

DP	DN	YP	YN	Duyarlılık	Kesinlik	F1 Skorlama
444	234	7	14	0,9694/0,9707	0,9844/0,9435	0,9768/0,9570

Duyarlılık, Kesinlik ve F1 Skorlama değerleri ilki iyi huylu ve sonraki kötü huylu tümör değerleri için verilmiştir.

Karmaşıklık matrisi ile elde edilen DP, DN, YP ve YN parametreleri kullanılarak modelin performansı belirlenmiştir. Algoritma performans ölçütlerinden duyarlılık, kesinlik ve F1 skorlama değerleri hem iyi huylu hem de kötü huylu tümör için ayrı ayrı hesaplanarak Tablo 4'te verilmektedir. Tablo 4 incelendiğinde modelin verileri doğru sınıflandırmada oldukça duyarlı olduğu yani iyi huylu sınıfına ait verilerin kendi sınıfına ve kötü huylu sınıfına ait verilerin kendi sınıfına yüksek oranda atadığı görülmektedir. Kesinlik ölçütüne bakıldığında iyi huylu verilerin sınıflandırılmasında yine oldukça yüksek bir oran elde dilmiştir. Yani pozitif değerler büyük ölçüde pozitif olarak belirlenmiştir. Ancak kötü huylu verilerin kötü olarak sınıflandırmada yani kötü huylu verilerin tespitinde modelin yeterince hassas olmadığı söylenebilir. F1 skoru yüksek olan Algoritmanın, homojen dağılımlı bir veri setindeki verileri yüksek başarı oranıyla sınıflandırıldığı ifade edilebilir.

3.2 Naive Bayes Sınıflandırıcı Uygulaması (Naive Bayes Classifier Implementation)

NB sınıflandırıcının uygulama verileri Tablo 5'te verilmiştir ve en iyi sonucu veren uygulamanın karmaşıklık matrisi Tablo 6'da sunulmuştur. NB sınıflandırıcıda useKernelEstimator ve useSupervisedDiscretization parametreleri öncelikle pasif durumda iken sınıflandırma yapılmış, ardından bu parametreler ayrı ayrı aktifleştirilerek sınıflandırma tekrar yapılmış ve doğrulama sonuçları buna göre bulunmuştur. Sınıflandırma sonuçlarına göre useKernelEstimator parametresinin aktif olduğu durumda elde edilen %97,42'lik doğrulama oranı en yüksek orandır. Yöntemin doğrulama sonuçları da Tablo 7'de verilmiştir.

Tablo 5. NB Sınıflandırıcının Sınıflandırma ve Doğrulama Sonuçları
(Classification and Validation Results of NB Classifier)

Sınıflandırıcı/Parametre	NB	NB[1*]	NB[2*]
Doğru Sınıflandırma	671	681	679
Yanlış Sınıflandırma	28	18	20
Doğrulama Oranı	95,99	97,42	97,14

[1]NB sınıflandırıcıda useKernelEstimator parametresi kullanılmıştır. [2]NB sınıflandırıcıda useSupervisedDiscretization parametresi kullanılmıştır.

Tablo 6. NB (useKernelEstimator) Sınıflandırıcının Karmaşıklık Matrisi
(NB (useKernelEstimator) Complexity Matrix of Classifier)

a	b	
442	16	a=2
2	239	b=4

* a veya 2 değeri iyi huylu tümörü, b değeri veya 4 kötü huylu tümörü temsil etmektedir.

Tablo 7. NB Sınıflandırıcının Performans Değerlendirme Ölçütleri
(Performance Evaluation Measures of NB Classifier)

DP	DN	YP	YN	Duyarlılık	Kesinlik	F1 Skorlama
442	239	2	16	0,9650/0,9917	0,9954/0, 9372	0,98/09637

Duyarlılık, Kesinlik ve F1 Skorlama değerleri iyi huylu ve kötü huylu tümör değerleri için verilmiştir.

Tablo 6'da NB sınıflandırıcının karmaşıklık matrisi DP, DN, YP ve YN parametreleri verilmektedir. Bu parametreler ile modelin performans ölçütleri olan duyarlılık, kesinlik ve F1 skorlama değerleri hem iyi huylu hem de kötü huylu tümör için ayrı ayrı hesaplanmıştır. Performans ölçütleri NB sınıflandırıcının bu veri seti için başarılı bir yöntem olduğunu göstermektedir. Ancak kötü huylu verilere yüksek oranda duyarlı (0,9917) olan modelin yine kötü huylu veriler için yüksek oranda hassas (0,9372) olmadığı söylenebilir. Ayrıca NB'nin F1 skoru verilerin homojen dağılıma sahip olduğunu göstermektedir.

3.3 J48 Karar Ağacı Sınıflandırıcı Uygulaması (J48 Decision Tree Classifier Implementation)

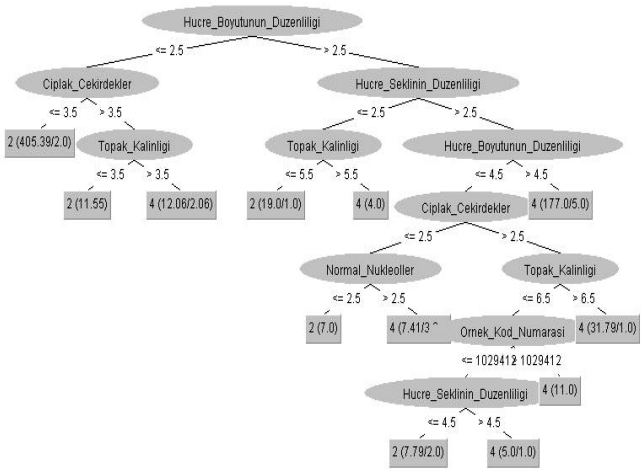
J48 algoritmasında bulunan Güven Faktörü (confidenceFactor) ve yaprak düğümünde bulunması gereken minimum obje sayısı (MinNumObj) parametrelerinin varsayılan değerleri değiştirilerek sınıflandırma başarısının artırılması hedeflenmiştir. Güven faktörü algoritmanın budama aşamasını kontrol etmektedir. Güven değeri bir yaprak düğümünde hata bulunma oranını ifade eder. Düğümde bulunan hata olasılığı güven değerinden fazla ise budanır. Düğümde varsayılan hata olasılıkla ifade edilmektedir. Dolayısıyla Güven değeri 0.5'ten küçük olma durumunda anlamlı olmaktadır. Güven değeri ne kadar az ayarlanırsa budama o kadar fazla olacaktır ve hataya karşı gösterilen tolerans o kadar az olmaktadır. Bu uygulamada Güven değeri 0,15, 0,20 ve 0,25 için, MinNumObj değeri ise 2, 3, 4, 5 değerleri için uygulanmıştır. Daha sonra en iyi sonucu veren doNotMakeSplitPointActualValue parametresi aktifleştirilmiş ve daha iyi bir sonuç için uygulamalar yapılmıştır. Tablo 8, farklı Güven Faktörü, MinNumObj ve doNotMakeSplitPointActualValue değerleri için J48 algoritmasının başarı oranlarını göstermektedir.

Tablo 8. J48 Sınıflandırıcı İçin Başarı Sonuçları
(Success Results for J48 Classifier)

Güven Faktörü / MinNumObj Değeri[1]*	2	3	4[2]*	5
0,15	94,99	95,28	95,28/95,28	95,28
0,20	95,13	94,99	95,42/95,42	95,13
0,25	94,56	95,13	95,42/95,56	94,99

1*Güven Faktörü değerleri 0,15-0,20 ve 0,25 olup MinNumObj Değerleri 2,3,4 ve 5'tir.
2* doNotMakeSplitPointActualValue parametresi kullanılmıştır.

Tablo 8'de görüldüğü üzere en iyi sonuç 0,25 Güven Faktörü değeri, MinNumObj=4 değeri ve doNotMakeSplitPointActualValue parametresinin "true" durumda olduğu sonuç için % 95,56 olarak verilmektedir. Algoritmanın sonuç olarak çıkardığı ağaç yapısı ise Şekil 3'te verilmiştir.



Şekil 3. J48 Ağaç Yapısı (J48 Tree Structure)

J48 algoritması veri kümesini Şekil 3'te verilen ağaç yapısı gibi sınıflandırmıştır. Ağaç irdelendiğinde en tepede Uniformity_of_cell_size özelliği, alt dallarda da diğer özelliklerin hangi sırada seçildiği ve bu kriterlere bağlı olarak tümör örneğinin iyi huylu/kötü huylu kararı gösterilmektedir. Bu seçim Shannon bilgi kuramından yararlanarak entropy hesabıyla seçilmektedir. Bu hesaba göre en iyi sonucu en tepedeki özellik yani Uniformity_of_cell_size vermektedir. Ağacın yapısına dikkat edildiğinde karmaşık ve geniş bir ağaç yerine kısa ve dar bir yapıda olmasından dolayı ağacın daha az dallanma ve daha az düğüm içerdiği ve bu ağaç yapısının performanslı olduğu söylenebilir.

J48 sınıflandırıcının en iyi sonucuna göre karmaşıklık matrisi değeri ise Tablo 9'da verilmektedir.

Tablo 9. J48 Sınıflandırıcı İçin Karmaşıklık Matrisi
(Complexity Matrix for J48 Classifier)

a	b	
441	17	a=2
14	227	b=4

* a değeri 2 yani iyi huylu tümörü, b değeri 4 yani kötü huylu tümörü temsil etmektedir.

Tablo 10. J48 Sınıflandırıcı için Performans Değerlendirme Ölçütleri
(Performance Evaluation Metrics for J48 Classifier)

DP	DN	YP	YN	Duyarlılık	Kesinlik	F1 Skorlama
441	227	14	17	0,9628/0,9419	0,9692/0,9303	0,966/0,936

Duyarlılık, Kesinlik ve F1 Skorlama değerleri iyi huylu ve kötü huylu tümör değerleri için verilmiştir.

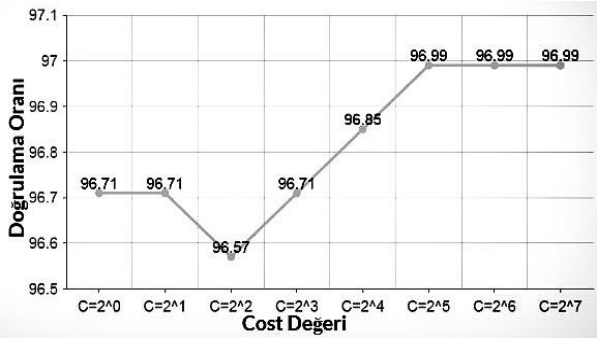
Tablo 9, J48 sınıflandırıcının karmaşıklık matrisini ve Tablo 10 ise modele ait hesaplanan duyarlılık, kesinlik ve F1 skorlama değerlerini göstermektedir. Performans ölçütleri modelin iyi huylu verileri sınıflandırmada daha başarılı olduğunu göstermektedir.

3.4 Destek Vektör Makinesi Sınıflandırıcı Uygulaması (Support Vector Machine Classifier Implementation)

DVM algoritması belirli parametrelerle Wisconsin veri setine uygulanarak elde edilen sonuçlar aşağıda verilmiştir. DVM uygulamasında C (Cost) değeri kritik bir parametre olup model performansı için logaritmik bir ölçekte geniş bir aralıktaki değerler ile denenir. Pratikte 2'nin katları şeklinde C değeri uygulamaları mevcuttur ancak zorunluluk değildir. Bu değerin yüksek olması modelin performansını artırmaktadır ancak ezberlemeye de (overfitting) yol açabilir. Bu çalışmada 1 (20) den 128 (27) e kadar C değeri denenmiştir ve Tablo 11'de DVM sınıflandırıcının doğruluk verileri elde edilmiştir. Sonuçlar Tablo 11 ve Şekil 4'te verilmiştir. İşlem yükü ve en iyi sonucun elde edildiği C değeri 678'dir.

Tablo 11. C Değerleri için DVM'nin Sınıflandırma ve Doğrulama Sonuçları
(Classification and Validation Results of SVM for C Values)

C Değerleri	Doğru Sınıflandırma	Yanlış Sınıflandırma	Doğrulama Oranı (%)
$C=2^0$	676	23	96,71
$C=2^1$	676	23	96,71
$C=2^2$	675	24	96,57
$C=2^3$	676	23	96,71
$C=2^4$	677	22	96,85
$C=2^5$	678	21	96,99
$C=2^6$	678	21	96,99
$C=2^7$	678	21	96,99



Şekil 4. DVM Doğrulama Grafiği (SVM Validation Chart)

Tablo 12. DVM Sınıflandırıcı için Karmaşıklık Matrisi
(Complexity Matrix for SVM Classifier)

a	b	
445	13	a=2
8	233	b=4

* a değeri 2 yani iyi huylu tümörü, b değeri 4 yani kötü huylu tümörü temsil etmektedir.

Tablo 13. DVM Sınıflandırıcı için Performans Değerlendirme Ölçütleri
(Performance Evaluation Metrics for SVM Classifier)

DP	DN	YP	YN	Duyarlılık	Kesinlik	F1 Skorlama
445	233	8	13	0,9716/0,9668	0,9823/0,9471	0,9769/0,9568

Duyarlılık, Kesinlik ve F1 Skorlama değerleri ilki iyi huylu ve sonraki kötü huylu tümör değerleri için verilmiştir.

Tablo 12, DVM için karmaşıklık matrisini ve Tablo 13'te ise algoritmanın performans ölçütlerini göstermektedir. Performans ölçütleri DVM'nin de iyi huylu verileri sınıflandırmada daha başarılı olduğunu göstermektedir. Diğer taraftan bazı kötü huylu tümör verileri tespit edemediği söylenebilir. Modelin başarı oranı, duyarlılık, hassasiyet ve F1 skoru parametrelerinin birbirine yakın olduğu görülmektedir. Çalışılan algoritmalar arasında DVM algoritması Yanlış Negatif parametresi en az olan bir algoritma olmuştur.

3.5 k-Means Kümeleme Algoritması Uygulaması (Application of k-Means Clustering Algorithm)

k-means kümeleme algoritmasında k değeri (numClusters) manuel olarak seçilir ve k 2, 3, 4, …n şeklinde değerler alabilir. Bu çalışmada kullanılan veri seti iki gruptan (iyi veya kötü huylu) oluştuğu için k =2 olarak seçilmiştir. Önemli bir diğer parametre ise uzaklık fonksiyonu olup bu çalışmada Öklid ve Manhattan uzaklık ölçütleri kullanılmıştır. Tablo 14'te k=2 için, Tablo 15'te k=3 için ve Tablo 16'da k=4 için model sonuçları verilmektedir. k=3 ve k=4 için sonuçların anlamsız olduğu görülmektedir. Bu durum iki grup/kümeden oluşan verilerin daha fazla kümeye ayrılması istendiğinde veriler arasındaki korelasyonun kaybolduğu görülmektedir. Küme sayısı bilinmeyen veri setlerinde denemeler veri setinin kaç kümeden oluştuğuna dair önemli birer yol göstericidir.

Tablo 14. k=2 İçin Kümeleme Sonuçları
(Clustering Results for k=2)

	Öklid Mesafe Fonksiyonu Örnek Sayısı – Yüzde		Manhattan Mesafe Fonksiyonu Örnek Sayısı - Yüzde	
Küme 0	246	35	237	34
Küme 1	453	65	462	66

Tablo 15. *k*=3 İçin Kümeleme Sonuçları
(Clustering Results for k=3)

	Öklid Mesafe Fonksiyonu Örnek Sayısı – Yüzde		Manhattan Mesafe Fonksiyonu Örnek Sayısı - Yüzde	
Küme 0	241	34	234	33
Küme 1	265	38	276	39
Küme 2	193	28	189	27

Tablo 16. *k*=4 İçin Kümeleme Sonuçları
(Clustering Results for k=2)

	Öklid Mesafe Fonksiyonu Örnek Sayısı – Yüzde		Manhattan Mesafe Fonksiyonu Örnek Sayısı - Yüzde	
Küme 0	239	34	136	19
Küme 1	17	2	114	16
Küme 2	188	27	188	27
Küme 3	255	36	261	37

Tüm bunların dışında sınıf sayısı adedince k değeri için yine farklı mesafe fonksiyonlarında "Sınıfların Kümelemelere Atanması (Classes to Clusters)" değerlendirmesi başarı yüzdeleri hesaplanabilmektedir. Doğrulama sonuçları Tablo 17'de verilirken karmaşıklık matrisi ise Tablo 18'de verilmiştir.

Tablo 17. *k*-Means Algoritması *k*=2 için Doğrulama Sonuçları (Classes to Clusters Evaluation)
(Validation Results for k-Means Algorithm k=2 (Classes to Clusters Evaluation))

	Doğru Sınıflandırma	Yanlış Sınıflandırma	Doğrulama Oranı(%)
Manhattan Fonksiyon	658	41	94,13
Öklid Fonksiyon	669	30	95,71

Tablo 18. *k*-Means Algoritması k=2 için Karmaşıklık Matrisi (Classes to Clusters Evaluation)
(Complexity Matrix for k-Means Algorithm k=2 (Classes to Clusters Evaluation))

Manhattan Fonksiyonu		Öklid Fonksiyonu		
a	**b**	**a**	**b**	
10	448	11	447	**a=2**
210	31	222	19	**b=4**

* a veya 2 değeri iyi huylu tümörü, b değeri veya 4 kötü huylu tümörü temsil etmektedir.

Bu iki tabloda da görüldüğü gibi k-means kümeleme algoritması ile elde edilen en yüksek doğruluk oranı %95,71 olmaktadır. Bu oran k-means algoritmasının Wisconsin verileri kümeleme sonuçlarının doğruluğu olup, modelin, verileri ayırmada kabul edilebilir başarıma sahip olduğu ifade edilebilir.

3.6 Hiyerarşik Kümeleme Algoritması Uygulaması (Implementation of Hierarchical Clustering Algorithm)

Hiyerarşik kümeleme algoritması k-means algoritmasındaki zorunlu olarak önceden girilen k adet küme parametresini belirleme işlemini ortadan kaldırmak için geliştirilen bir algoritmadır. k-means'te kümeleme işlemi, önceden belirlenmiş küme sayısına göre belirlenen küme merkezleri oluşturularak veri noktalarını bu merkezlere olan yakınlıklarına göre yapar. Kümeler ortalama değerlerine göre sürekli güncellenir. Hiyerarşik kümelemede ise her bir veri noktasını bir küme kabul eder (Agglomerative yöntemi) ve en yakın iki veri noktasını birleştirerek yeni kümeler oluşturur. Daha sonra bu kümeler yakınlıklarına göre birleştirilir ve bu işlem kümeleme işlemi bitene kadar devam eder. Hiyerarşik kümelemedeki k parametresi dendrogram üzerinde hangi seviyede kesme yapılacağını belirler. Literatürde bu veri seti için hiyerarşik metodu önerilen algoritmalar arasında yer almamaktadır. Bu çalışmada modelin neden Wisconsin verilerini ayırmada başarılı olmadığı noktası ele alınmıştır. Kümeleme yöntemleri verilerin benzerlik durumlarını veriler arasındaki mesafe ile ilişkilendirilmektedir. İki veri arasındaki mesafe ölçme metodu önem kazanmaktadır. Hiyerarşik kümeleme yöntemi Wisconsin veri seti için önemli bir örnek teşkil etmektedir. k=2 değeri ve farklı mesafe fonksiyonları için yapılan uygulamada elde edilen sonuçlar Tablo 19'da yer almaktadır.

Tablo 19. Hiyerarşik Kümeleme k=2 İçin Kümeleme Sonuçları
(Clustering Results for Hierarchical Clustering k=2)

	Öklid Mesafe Fonksiyonu Örnek Sayısı – Yüzde	Manhattan Mesafe Fonksiyonu Örnek Sayısı - Yüzde
Küme 0	**458 - 66**	**698 - 100**
Küme 1	**241 - 34**	**1 - 0**

Bu tabloda özellikle Manhattan Fonksiyonu kullanıldığında kümelemedeki dengesiz dağılım açık şekilde görülmektedir. Özellikle kötü huylu tümör verilerini ayırmada oldukça başarısız olduğu görülmektedir. Model veriler arasındaki uzaklık bilgisine göre kümeleme yapmaktadır. Yüksek boyutlu veri setlerinde Hiyerarşik yöntemi, verilerin birbiri ile benzerliği ve uzaklığını tespit etmede zorlanabilmektedir. Yakın olan veriler benzer olmakta ve benzer olan örnekler aynı grupta değerlendirilmektedir. Dolayısıyla uzaklık

fonksiyonu Hiyerarşik kümeleme algoritması için oldukça önemli bir parametre olmaktadır. Manhattan uzaklık tekniği yatay veya dikey birim adım şeklinde iki veri arasındaki uzaklığı belirlemektedir. Wisconsin veri seti gibi çok boyutlu veri setlerinde bu ölçüm tekniği verileri ayırmada başarısız olmaktadır.

Hiyerarşik kümeleme algoritması farklı mesafe fonksiyonlarında "Sınıfların Kümelemelere Atanma (Classes to Clusters)" başarı yüzdeleri hesaplanabilmektedir. Doğrulama sonuçları Tablo 20'de verilirken karmaşıklık matrisi parametreleri ise Tablo 21'de verilmiştir.

Tablo 20. Hiyerarşik Kümeleme Algoritması Doğrulama Sonuçları
(Hierarchical Clustering Algorithm Validation Results)

	Doğru Sınıflandırma	**Yanlış Sınıflandırma**	**Doğrulama Oranı(%)**
Manhattan Fonksiyonu	459	240	65,66
Öklid Fonksiyonu	459	240	65,66

Tablo 21. Hiyerarşik Kümeleme Algoritması Karmaşıklık Matrisi
(Hierarchical Clustering Algorithm Complexity Matrix)

Manhattan Fonksiyonu		**Öklid Fonksiyonu**		
a	**b**	**a**	**b**	
458	0	458	0	**a=2**
240	1	240	1	**b=4**

* a veya 2 değeri iyi huylu tümörü, b değeri veya 4 kötü huylu tümörü temsil etmektedir.

Bu iki Tabloda da görüldüğü gibi Hiyerarşik kümeleme algoritmasında en yüksek doğruluk oranı %65,66 olmaktadır. Ayrıca karmaşıklık matrisinde de görüleceği üzere a değeri yani iyi huylu tümörler başarılı şekilde kümelendirilirken, b yani kötü huylu tümörler kümelendirmede model neredeyse tamamen başarısız olmaktadır.

4. TARTIŞMA (DISCUSSION)

Bu çalışmada, k-NN, NB ve DVM sınıflandırma algoritmaları ile k-means ve Hiyerarşik kümeleme algoritmaları, Weka 3.8.6 makine öğrenimi aracı kullanılarak Wisconsin meme kanseri veri setine uygulanmış ve elde edilen sonuçlar tablolarda sunulmuştur.

İlk olarak, k-NN sınıflandırıcısı veri setine uygulanarak optimal bir k-değeri araştırılmıştır. Yapılan uygulamalarda k=5 değeri için %96,99 doğruluk derecesine sahip bir başarı derecesi elde edilmiştir. Modelin performansı analiz edildiğinde k=5 için k-NN algoritması, 458 iyi huylu tümörün (benign) 444'ünü ve 241 kötü huylu tümörün (malign) 234'ünü doğru bir şekilde sınıflandırmıştır. Bu değerler karmaşıklık matrisi ile analiz edildiğinde k-NN modellinin oldukça hassas ve duyarlı olduğu görülmekle birlikte etkili performansıyla yüksek başarıma sahip olduğu söylenebilir. Diğer taraftan çok kısa inşa sürelerine sahip hafif, tembel bir öğrenme algoritması olduğu göz önüne alındığında, modelin yüksek doğruluk derecesi ile diğer sınıflandırıcılara göre Wisconsin verilerini sınıflandırmada iyi olduğu söylenebilir. Ayrıca, karmaşıklık matrisi kullanılarak performans değerlendirme ölçütleri hesaplanmıştır. Buna göre hem iyi huylu hem de kötü huylu tümörün sınıflandırılmasında performans ölçütleri olumlu sonuçlar vermektedir. Duyarlılık parametresinde kötü huylu tümörün iyi huylu tümöre göre az bir farkla daha doğru tahmin edildiği; kesinlik parametresinde ise iyi huylu olarak sınıflandırılmış olan örneklerin oranının kötü huylu tümöre göre daha doğru tahmin edildiği görülmektedir. Ayrıca F1 skorlama parametresinde her iki tümör de yüksek oranda doğru tahmin edilmiştir. Ancak iyi huylu tümörlerin sınıflandırılmaları daha başarılı bir şekilde yapılmıştır.

Bu çalışmada kullanılan bir diğer sınıflandırıcı olan NB'nin doğruluk oranı en iyi durumda (useKernelEstimator aktif iken) %97,42'dir. 458 iyi huylu tümör vakasının 446'sını ve 241 kötü huylu tümör vakasının 239'unu doğru bir şekilde sınıflandırabilmiştir. Bu yöntem, Weka programında useKernelEstimator ve useSupervisedDiscretization parametreleri kullanılarak incelenmiş ve %97,42 doğruluk oranı ile en iyi sonuç olarak değerlendirilmektedir. Karmaşıklık matrisi kullanılarak performans değerlendirme ölçütleri hesaplandığında hem iyi huylu hem de kötü huylu tümörün sınıflandırılmasında performans ölçütleri olumlu sonuçlar vermektedir. Duyarlılık parametresinde kötü huylu tümörün iyi huylu tümöre göre daha doğru tahmin edildiği; kesinlik parametresinde ise iyi huylu olarak sınıflandırılmış olan örneklerin oranının kötü huylu tümöre göre çok daha doğru tahmin edildiği görülmektedir. F1 skorlama parametresinde ise her iki tümörün de iyi oranda tahmin edildiği görülmekte birlikte modelin iyi huylu tümörü sınıflandırmada daha başarılı olduğu tespit edilmiştir. NB sınıflandırıcı iyi huylu tümörde %98 oran, kötü huylu tümörde ise %96,37 oran ve en iyi F1 skorunu vermektedir.

J48 algoritmasında Güven Faktörü (confidenceFactor), MinNumObj ve doNotMakeSplitPointActualValue parametrelerinin doğruluk üzerindeki etkisi üzerinde durulmuştur. ConfidenceFactor=0,25, MinNumObj=4 ve doNotMakeSplitPointActualValue parametresi "true" durumda iken en optimum sonuç elde edilmiş ve doğrulama yüzdesi %95,56 olmuştur. Modelin karmaşıklık matrisi duyarlılık ve kesinlik parametreleri iyi huylu tümörün kötü huylu tümöre göre daha doğru tahmin edildiğini belirtmektedir. F1 skorlama parametresinde her iki tümörün de iyi oranda tahmin edildiği göstermektedir. Ancak model iyi huylu tümörü sınıflandırmada daha başarılı olduğu sonucuna varılmıştır.

DVM algoritmalarının birçok alt-sınıflandırıcısı ve değiştirilebilecek birçok parametresi bulunmaktadır. Bu çalışmada, maliyet (C) parametresinin doğruluk üzerindeki

etkisi üzerinde durulmuştur. Sunulan sonuçlar göz önüne alındığında, en iyi sonuç C=25 değeri için %96,99 olarak elde edilmiştir ve bu sınıflandırıcı, k-NN sınıflandırıcısı ile aynı doğruluk oranına sahip olup kabul edilebilir bir doğruluk oranıdır. Karmaşıklık matrisi duyarlılık ve kesinlik parametreleri iyi huylu tümörlerin sınıflandırılmasında modelin daha başarılı olduğunun göstermektedir. F1 skorlama ölçütü modelin tümör tespitinde başarılı olduğu ve iyi huylu tümör tespitinde daha çok başarılı olduğunu göstermektedir.

Kümeleme algoritmalarından k-means ile belli mesafe fonksiyonları ve k sayısının (numClusters) kümeleme üzerindeki etkisi üzerinde durulmuştur. Aynı zamanda k=2 değeri ve Öklid mesafe fonksiyonu değeri içinde doğruluk oranı belirlenmiştir. Bu uygulamada k-means için %95,71 doğruluk oranı belirlenmiştir.

Diğer bir kümeleme algoritması olan Hiyerarşik kümeleme Wisconsin veri setine uygulanmıştır. Bu uygulamada da belli mesafe fonksiyonları ve k sayısının (numClusters) kümeleme üzerindeki etkisi üzerinde durulmuştur. Burada da k=2 değeri ve Öklid mesafe fonksiyonu değeri içinde doğruluk oranı belirlenmiştir. Bu durumda %65,66'lık bir doğrulama oranı oluşmuştur. Bu oran kümeleme algoritmasının bu veri setinde başarısız olduğu açık bir göstergesidir. Diğer taraftan Manhattan uzaklık tekniği seçildiğinde yöntemin bu veri setinde tamamen başarısız olduğu gözlemlenmiştir. Dolayısıyla Wisconsin verilerini gruplara ayırırken uzaklık fonksiyonunun önemli bir parametre olduğu görülmektedir. Verilerin benzerliği, iki verinin birbirine yakınlığı üzerine kurulu olduğu için uzaklık ölçütü modellerin performansında önemli rol oynamaktadır. Modellerin doğrulama oranları Tablo 22'de verilmiştir.

Tablo 22. Sınıflandırma ve Kümeleme Algoritmalarının Doğrulama Yüzdeleri
(Validation Percentages of Classification and Clustering Algorithms)

Algoritma Adı	Doğrulama Oranı(%)
k-NN	96,99
NB	97,42
J48	95,56
DVM	96,99
k-means	95,71
Hiyerarşik	65,66

Bu çalışmanın, literatürdeki benzer çalışmalarla doğruluk oranı karşılaştırma tablosu (Tablo 23) aşağıda verilmektedir.

Tablo 23. Algoritmaların Doğrulama Yüzdeleri Literatür Karşılaştırmaları
(Validation Percentages of Algorithms Literature Comparisons)

Çalışmalar	DVM	k-NN	J48	NB
Amrane vd. 2018 [7]	-	97,51	-	96,19
Aruna vd. 2011 [8]	96,84	-	94,59	96,50
Akbugday 2019 [33]	96,85	96,85	-	95,99
Ahmed vd. 2020 [32]	96,13	-	94,26	97,27
Uddin vd. 2024 [9]	90,15	89,63	91,21	-
Nemade vd. 2023 [10]	95	96	97	90
Kadhim vd. 2022 [14]	96,49	95,61	-	91,22
Laghmati vd. 2023 [12]	92,1	93,8	-	-
Amethiya vd. 2021 [13]	94,3	95,9	94,56	-
Bu çalışma	**96,99**	**96,99**	**95,56**	**97,42**

Tablo 23'de görüldüğü üzere bu çalışma, benzer çalışmalarla karşılaştırıldığında tutarlı olduğu görülmüş ve literatürden dahi iyi sonuçlar elde edilmiştir.

Modellerin hem iyi huylu hem de kötü huylu tümör için F1 skorlamaları ve ROC sonuçları Tablo 24'te verilmiştir.

Tablo 24. Sınıflandırma Algoritmalarının F1 Skorlama Yüzdeleri
(F1 Scoring Percentages of Classification Algorithms)

Algoritma Adı	F1 Skorları(%) (İyi Huylu/ Kötü Huylu)	ROC
k-NN	97,68 / 95,70	0,970
NB	98 / 96,37	0,978
J48	96,6 / 93,6	0,952
DVM	97,69 / 95,68	0,969

Meme kanseri veri setine uygulanan algoritma başarılarını karşılaştırmak için ROC analiz yapılmış modellerin başarıları sırayla NB (0,978), k-NN (0,970), DVM (0,969), j48 (0,952), k-means (0,948) ve Hiyerarşik (0,502) olarak elde edilmiştir. ROC kriteri modellerin başarılarını karşılaştırmak için kullanılan etkili bir kriterdir. Modellerin ROC sonuçları sınıflandırıcıların kümeleme yöntemlerine göre daha başarılı olduğunu göstermektedir. Sınıflandırıcılar arasında da en başarılı yöntem NB algoritması olmuştur. ROC sonuçları modellerin başarı oranı, duyarlılık, keskinlik ve F1 skorlama kriterleri ile uyumludur.

5. SONUÇ VE ÖNERİLER (RESULT AND SUGGESTIONS)

İnsan yaşam kalitesini artırmaya yönelik yapılan çalışmalar giderek yoğunlaşmaktadır. Veri bilimindeki gelişmeler, verilerin farklı modellerle analiz edilerek çeşitli yönleriyle değerlendirilmesi ve daha başarılı sonuçlara ulaşılmasını sağlamaktadır. Bu çalışmada, kadınlar arasında oldukça yaygın ve ciddi olan meme kanseri için otomatik tanı sistemi araştırılmıştır. Çeşitli sınıflandırma ve kümeleme yöntemleri, UCI veri tabanından alınan meme kanseri veri setine uygulanarak modellerin performansları farklı metriklerle değerlendirilmiştir. Veri seti analiz edilerek, hastalık tespitinde etkin özellikler belirlenmiştir. Sürekli artan verilerden bilgi çıkarma veya veri madenciliği süreci detaylı olarak ele alınmıştır. Algoritmaların sonuçları, geleneksel başarı oranı metrikleri üzerinden değil, daha detaylı ve yeni bir görüş olan her bir verinin kendi sınıfına aitliği tespiti de irdelenmiştir. Oldukça karmaşık bir yapıya sahip olan derin öğrenme ile de hastalık tespiti yapılabilir; ancak bu çalışmada model performanslarının daha detaylı analizi mümkün olduğu düşünülmektedir. Benzer şekilde, literatürde bu veri seti için modellerin başarı oranı üzerinde durulurken, bu çalışmada başarılı bir tanı için önemli metrikler analiz edilmiştir.

Çalışmada k-NN, NB, J48 ve DVM sınıflandırma algoritmaları ile k-means ve hiyerarşik kümeleme algoritmaları kullanılmıştır. Sonuçlar, %97,42 doğruluk oranıyla NB algoritmasının en doğru sınıflandırma algoritması olduğunu göstermektedir. Ayrıca, iyi huylu tümör için %98 ve kötü huylu tümör için %96,37 oranıyla en iyi F1 skorlamasına sahip sınıflandırıcı yine NB algoritması olmuştur. Modelin ROC değeri de sonuçlarla uyumludur.

Veri setinde yer alan örneklerin özellikleri, başarılı tanı için önemli bir kriterdir. İki örnek benzerliği değerlendirilirken kullanılan mesafe ölçütleri (Öklid, Manhattan, Minkowski mesafe fonksiyonları) kritik öneme sahiptir. İki veri arasındaki mesafe, verilerin benzerliklerini göstermekte ve başarılı teşhis ile doğrudan orantılıdır.

Günümüzde ön tanı veya karar destek sistemlerinin günlük hayatta kullanılabilme imkanları mevcuttur. Yeni mobil cihazların kapasitesi oldukça gelişmiş olup, birçok marka kendi cihazlarına yapay zekâ uygulama programlama arayüzlerini (Application Programming Interface-API) entegre etmeye başlamışlardır. Bu minvalde, hastalıkların tespiti için kullanılan test cihazlarına, o hastalık tanısında başarılı olan algoritmaların entegre edilmesiyle ön tanı işlemleri mümkün hale gelebilir. Örneğin, başka bir şikâyet üzerine yapılan test sonuçlarının otomatik algoritmalar tarafından değerlendirilerek varsa başka hastalıkların da ortaya çıkarılması söz konusudur. Böyle bir sistem ile hastalığın önceden tespiti mümkün olabilir. Bu tür sistemler, hekimlerin aşırı iş yükü gibi çeşitli faktörlerden dolayı hastalığın farkına varamaması ihtimalini düşürür ve insan kaynaklı hataları minimize edebilir.

Gelecekteki çalışmalarda, farklı veri setleri veya farklı parametreler içeren veriler ile yapay zekâ konusunda daha gelişmiş programlama dilleri kullanılarak algoritmaların farklı platformlardaki davranışı araştırılabilir. Gelişmiş programlama uygulamaları ve/veya platforma özgü avantajlarla daha doğru sınıflandırıcıların kullanımı, daha kesin sonuçların elde edilmesini sağlayabilir.

KAYNAKLAR (REFERENCES)

[1] Siegel, R.L., Miller, K.D. and Jemal, A., 2020. Cancer statistics, 2020. CA: *a cancer journal for clinicians*, 70(1), 7–30. https://doi.org/10.3322/caac.21590.

[2] Bora B., Soytutan Ç.İ., Aygün A., Özdemir T.A., Kulali B., Uzun S.B. ve ark., (2019). Sağlık İstatistikleri Yıllığı. Sağlıkta İstatistik ve Nedensel Analizler (SİNA) Platformu. Ankara, Türkiye.

[3] Anderson, Benjamin O., et al., 2010. Optimisation of breast cancer management in low-resource and middle-resource countries: executive summary of the Breast Health Global Initiative consensus. The lancet oncology, 12.4 (2011): 387-398.

[4] Internet: Bakanlığı, T. S. (2022). Sağlık istatistikleri yıllığı. Türkiye İstatistik Kurumu (TUİK). Ankara.

[5] Türkyılmaz, M., Öztürk, M., Dündar, S., Ergün, K.A., Sevinç, A., Tütüncü, S., Seymen, E., (2021). Türkiye Kanser İstatistikleri. T.C. Sağlık Bakanlığı Halk Sağlığı Genel Müdürlüğü. Ankara, Türkiye.

[6] Internet: UC Irvine, Breast Cancer Wisconsin (Original), UC Irvine Machine Learning Repository. https://archive.ics.uci.edu/dataset/15/breast+cancer+wisconsin+original, (14.07.1992)

[7] Amrane, M., Oukid, S., Gagaoua, I., and Ensari, T., 2018. Breast cancer classification using machine learning. In 2018 electric electronics, computer science, biomedical engineerings' meeting (EBBT) (pp. 1-4). IEEE.

[8] Aruna, S., S. P. Rajagopalan, and L. V. Nandakishore, 2011. Knowledge based analysis of various statistical tools in detecting breast cancer. Computer Science & Information Technology, 2.2011 (2011): 37-45.

[9] Uddin, K. M. M., Biswas, N., Rikta, S. T., & Dey, S. K. (2023). Machine learning-based diagnosis of breast cancer utilizing feature optimization technique. Computer Methods and Programs in Biomedicine Update, 3, 100098.

[10] Nemade, V., & Fegade, V. (2023). Machine learning techniques for breast cancer prediction. Procedia Computer Science, 218, 1314-1320.

[11] Singh, L. K., Khanna, M., & Singh, R. (2024). An enhanced soft-computing based strategy for efficient feature selection for timely breast cancer prediction: Wisconsin Diagnostic Breast Cancer dataset case. Multimedia Tools and Applications, 1-66.

[12] Laghmati, Sara & Hamida, Soufiane & Hicham, Khadija & Cherradi, Bouchaib & Tmiri, Amal. (2023). An improved breast cancer disease prediction system using ML and PCA. Multimedia Tools and Applications. 83. 1-37. 10.1007/s11042-023-16874-w.

[13] Amethiya, Yash & Pipariya, Prince & Patel, Shlok & Shah, Manan. (2021). Comparative Analysis of Breast Cancer detection using Machine Learning and Biosensors. Intelligent Medicine. 2. 10.1016/j.imed.2021.08.004.

[14] Kadhim, Rania & Kamil, Mohammed. (2022). Comparison of breast cancer classification models on Wisconsin dataset. *International Journal of Reconfigurable and Embedded Systems (IJRES)*. 11. 166-174. 10.11591/ijres.v11.i2.pp166-174.

[15] Internet: Öğüdücü, Ş.G., Veri Madenciliği Temel Sınıflandırma Yöntemleri, https://web.itu.edu.tr/~sgunduz/courses/verimaden/slides/d3.pdf.

[16] Uğuz, S. (2019). Makine öğrenmesi teorik yönleri ve Python uygulamaları ile bir yapay zekâ ekolü. Nobel Yayıncılık. Ankara.

[17] Internet: Akçay, A. K En Yakın Komşu Algoritması. https://aycaakcay.medium.com/k-en-yakin-komsu-k-nearest-neighbour-algoritmasi-siniflama-7c456f8e2b0d, (25.06.2020).

[18] Internet: Şeker, Ş.E., KNN(K Nearest Neighborhood, En Yakın k Komşu). https://bilgisayarkavramlari.com/2008/11/17/knn-k-nearest- neighborhood-en-yakin-k-komsu/, (17.11.2008).

[19] Internet: Hatipoğlu, E. Machine Learning – Classification – Naive Bayes – Part 11. https://medium.com/@ekrem.hatipoglu/machine-learning-classification-naive-bayes-part-11-4a10cd3452b4, (13.06.2018).

[20] Solmaz, R., Günay, M., and Alkan, A., (2014). Fonksiyonel Tiroit Hastalığı Tanısında Naive Bayes Sınıflandırıcının Kullanılması. Akademik Bilişim Konferansı. Mersin, Türkiye, 891-897.

[21] Hemanth, D. J., and Kose, U., 2020. Artificial Intelligence and Applied Mathematics in Engineering Problems: Proceedings of the International Conference on Artificial Intelligence and Applied Mathematics in Engineering (ICAIAME 2019). Vol. 43,Springer Nature.

[22] Internet: Medium Yöntemler – 4.1: C4.5 Algoritması, https://medium.com/@Emreyz/yontemler-4-1- c4-5-algoritmasi-7382de92584e, (03.03.2017).

[23] Internet: Şeker, Ş.E., C4.5 Ağacı, https://bilgisayarkavramlari.com/2012/11/13/c4-5-agaci-c4-5-tree/, (13.11.2012).

[24] Aras, Ü., 2008. **Finansal veri madenciliği**. Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, 85.

[25] Internet: Şeker, Ş.E., SVM (Support Vector Machine, Destekçi Vektör Makinesi), https://bilgisayarkavramlari.com/2008/12/01/svm-support-vector-machine-destekci-vektor-makinesi/, (01.12.2008).

[26] Internet: Şeker, Ş.E., Weka ile SVM, https://bilgisayarkavramlari.com/2011/09/19/weka-ile-svm/, (19.09.2011).

[27] Solmaz, R., Günay, M., and Alkan, A., 2013. Uzman sistemlerin tiroit teşhisinde kullanılması. XV. Akademik Bilişim Konferansı Bildirileri, 23-25.

[28] Internet: Çalışkan, T.K., Destek Vektör Makineleri (DVM), https://www.bilimma.com/destek-vektor-makineleri-support-vectors-machines-svms, (01.04.2023).

[29] Internet: Şeker, Ş.E., K-Ortalama Algoritması (K-Means Algorithm), https://bilgisayarkavramlari.com/2008/12/15/k-ortalama-algoritmasi-k-means-algorithm/, (15.12.2008).

[30] Takaoğlu, M., and Takaoğlu, F., 2019. K-Means ve Hiyerarşik Kümeleme Algoritmanın Weka ve Matlab Platformlarında Karşılaştırılması. İstanbul Aydın Üniversitesi Dergisi, 11(3), 303-317.

[31] Internet: Seker, S. E., 2015. Sosyal ağlarda veri madenciliği (data mining on social networks). Ybs Ansiklopedi, 2.2 (2015): 30-39.

[32] Ahmed, M. T., Imtiaz, M. N., and Karmakar, A., 2020. Analysis of wisconsin breast cancer original dataset using data mining and machine learning algorithms for breast cancer prediction. Journal of Science Technology and Environment Informatics, 9(2), 665-672

[33] Akbugday, B., 2019. Classification of breast cancer data using machine learning algorithms. In 2019 Medical technologies congress (TIPTEKNO) (pp. 1-4). IEEE