T.C.

YALOVA ÜNİVERSİTESİ

MÜHENDİSLİK FAKÜLTESİ

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

MAKİNE ÖĞRENMESİ DERSİ VİZE PROJESİ

KRONİK BÖBREK HASTALIĞI TEŞHİSİNİ MAKİNE ÖĞRENMESİ ALGORTİMALARI İLE SINIFLANDIRMA

HAZIRLAYANLAR

200101062 ÖZLEM AKINCI

200101069 ÇAĞLA ÖKMEN

DANIŞMAN : MURAT GÖK

YALOVA-2025

İçindekiler

Şekiller Listesi

Tablolar Listesi

Grafik Listesi

Kısaltma Listesi

CKD: Chronic Kidney Disease (Kronik Böbrek Hastalığı)

UCI: University of Irvine (Irvine Üniversitesi)

ML: Machine Learning (Makine Öğrenmesi)

PCA: Principal Component Analysis (Temel Bileşenler Analizi)

GA: Genetic Algorithm (Genetik Algoritma)

KBH: Kronik Böbrek Hastalığı

ckd: Chronic Kidney Disease (Kronik Böbrek Hastası)

notckd: Not Chronic Kidney Disease (Kronik Böbrek Hastalığı Değil)

VM: Veri Madenciliği

MV: Missing Value (Eksik Değer)

KNN: K-Nearest Neighbors (K-En Yakın Komşu)

SVM: Support Vector Machine (Destek Vektör Makinesi)

RBF: Radial Basis Function (Radyal Tabanlı Fonksiyon)

MLP: Multi-Layer Perceptron (Çok Katmanlı Algılayıcı)

MCC: Matthews Correlation Coefficient (Matthews Korelasyon Katsayısı)

TP: True Positive (Gerçek Pozitif)

TN: True Negative (Gerçek Negatif)

FP: False Positive (Yanlış Pozitif)

FN: False Negative (Yanlış Negatif)

P: Positive (Pozitif)

N: Negative (Negatif)

KRONİK BÖBREK HASTALIĞI TEŞHİSİNİ MAKİNE ÖĞRENMESİ ALGORTİMALARI İLE SINIFLANDIRMA

Abstarct

This study aims to accurately diagnose individuals with chronic kidney disease (CKD). CKD is a significant global public health issue, and its early diagnosis is challenging. To address this, the performance of various machine learning algorithms was compared using an open-source dataset obtained from the University of California Irvine (UCI) machine learning (ML) repository. Nine different machine learning algorithms were evaluated using 10-fold cross-validation, feature selection-based genetic algorithms, and PCA for dimensionality reduction. The dataset comprises 400 samples and 24 attributes. The primary objective of the study is to identify the prediction model with the highest accuracy rate, thereby determining effective algorithms for CKD diagnosis.

Keywords: Chronic Kidney Disease, Machine Learning, Classification, Genetic Algorithms (GA), Principal Component Analysis (PCA)

Özet

Bu çalışma, kronik böbrek hastalığı (KBH) tanısı almış bireylerin doğru teşhisini amaçlamaktadır. KBH, dünya genelinde önemli bir halk sağlığı sorunudur ve erken teşhisi zordur. Bu amaçla, University of California Irvine makine öğrenmesi deposundan elde edilen açık kaynaklı bir veri seti kullanılarak farklı makine öğrenmesi algoritmalarının performansı karşılaştırılmıştır. Çalışmada, 10 katlı çapraz doğrulama, öznitelik seçimine bağlı genetik algoritmalar ve PCA ile özellik indirgeme yöntemleri uygulanarak 9 farklı makine öğrenmesi algoritması değerlendirilmiştir. Veri seti, 400 örnek ve 24 öznitelik içermektedir. Çalışmanın temel amacı, en yüksek doğruluk oranına sahip tahmin modelini belirleyerek KBH teşhisinde kullanılabilecek etkili algoritmaları belirlemektir.

Anahtar Kelimler: Kronik Böbrek Hastalığı, Makine Öğrenmesi, Sınıflandırma, Genetik Algortimalar, PCA

1 Giriş

Sağlık, bireylerin yaşamını doğrudan etkileyen en temel unsurdur. Hastalıklar, doğru fark edilmediğinde ciddi sağlık sorunlarına ve gecikmelere yol açabilmektedir. Bu hastalıklardan biri olan kronik böbrek hastalığı, dünya genelinde milyonlarca insanı etkileyen yaygın ve önemli bir sağlık problemidir. Giderek artan bu halk sağlığı sorunu dünya genelinde yaygınlığının %8-16 arasında olduğu tahmin edilmektedir [1].

KBH, dünya nüfusunun yaklaşık %10'unu etkileyen küresel bir halk sağlığı sorunudur [2] [3]. Çin'de KBH prevalansı oranı %10,8'dir [4] ve Amerika Birleşik Devletleri'nde prevalans aralığı %10-%15'tir [5]. Başka bir çalışmaya göre, bu oran Meksika yetişkin genel popülasyonunda %14,7'ye ulaşmıştır [6]. Bu hastalık, böbrek fonksiyonlarında yavaş bir bozulma ile karakterize edilir ve bu da nihayetinde böbrek fonksiyonlarının tamamen kaybına neden olur. KBH, erken evrelerinde belirgin semptomlar göstermez. Bu nedenle hastalık, böbrek fonksiyonlarının yaklaşık %25'ini kaybedene kadar fark edilmeyebilir [7]. Erken belirti göstermemesi ve yaygın görülmesi, KBH'nin teşhisini daha da kritik bir hâle getirebilir.

Bu çalışmada ise kronik böbrek hastalığına sahip bireylerin teşhisinin doğru tahminini bulmak amacıyla farklı makine öğrenmesi algoritmaları karşılaştırmalı olarak uygulanacaktır. Açık kaynaklı olarak Kaggle üzerinden alınan "Kidney Disease" veri seti üzerinde çeşitli sınıflandırma modelleri test edilecek ve performansları değerlendirilecektir. Çalışmanın temel amacı, kullanılan makine öğrenmesi algoritmaları arasından en yüksek doğruluk oranına sahip tahmin modelini belirlemektir. Elde edilen sonuçlar, böbrek hastalığı tespiti konusunda hangi algoritmaların daha başarılı olduğunu ortaya koyacaktır.

2 Yöntem

Bu çalışmada, kronik böbrek hastalığı teşhisinin doğruluğunu kontrol etmek amacıyla veri setimize üç farklı yöntem (10 katlı çapraz doğrulama, Öznitelik seçimine bağlı Genetik Algortimalar ve PCA ile özellik daraltma uygulanarak) altında 9 farklı makine öğrenmesi algoritmalarının performansları karşılaştırılacaktır.

2.1 Veri Seti

Bu çalışmada Kronik Böbrek Hastalığı tahmini için University of California Irvine Machine Learning Repository'de depolanmış "Kidney Disease" veri setinin, açık kaynaklı Kaggle platformu üzerinden Mansoor Iqbal tarafından [8] yayımlanmış olan versiyonu kullanılmıştır.

Veri seti 400 örnek içermektedir. Her bir örnekte 24 tahmin değişkeni bulunmaktadır. Bunların 11'i sayısal (numerical), 13'ü kategorik (nominal) veridir. Ayrıca veri setinde bir adet kategorik yanıt (output) değişkeni(class) yer almaktadır. Sınıfın iki değeri vardır: ckd (KBH olan örnekler) ve notckd (KBH olmayan örnekler).

400 örnekte, **250 örnek ckd kategorisine, 150 örnek ise notckd kategorisine aittir**. Veri setinde (Tablo 1'de gösterildiği gibi) **çok sayıda eksik değer** bulunmaktadır [2].

Variable	Açıklama	Tür (Class)	Ölçek (Scale)	Eksik Veri Oranı
age	Age	Numerical	age in years	2.25%
bp	Blood Pressure	Numerical	in mm/Hg	3%
sg	Specific Gravity	Nominal	(1.005, 1.010, 1.015, 1.020, 1.025)	11.75%
al	Albumin	Nominal	(0, 1, 2, 3, 4, 5)	11.5%
su	Sugar	Nominal	(0, 1, 2, 3, 4, 5)	12.25%
rbc	Red Blood Cells	Nominal	(normal, abnormal)	38%
pc	Pus Cell	Nominal	(normal, abnormal)	16.25%
pcc	Pus Cell Clumps	Nominal	(present, notpresent)	1%
ba	Bacteria	Nominal	(present, notpresent)	1%
bgr	Blood Glucose Random	Numerical	in mgs/dl	11%
bu	Blood Urea	Numerical	in mgs/dl	4.75%
sc	Serum Creatinine	Numerical	in mgs/dl	4.25%
sod	Sodium	Numerical	in mEq/L	21.75%
pot	Potassium	Numerical	in mEq/L	22%
hemo	Hemoglobin	Numerical	in gms	13%
pcv	Packed Cell Volume	Numerical	-	17.75%
wbcc	White Blood Cell Count	Numerical	in cells/cumm	26.5%
rbcc	Red Blood Cell Count	Numerical	in millions/cmm	32.75%
htn	Hypertension	Nominal	(yes, no)	0.5%
dm	Diabetes Mellitus	Nominal	(yes, no)	0.5%
cad	Coronary Artery Disease	Nominal	(yes, no)	0.5%
appet	Appetite	Nominal	(good, poor)	0.25%
pe	Pedal Edema	Nominal	(yes, no)	0.25%
ane	Anemia	Nominal	(yes, no)	0.25%
class	Class (hedef değişken)	Nominal	(ckd, notckd)	0%

Tablo 1:Kidney Disease Veri Seti Değişkenleri

2.2 Veri Ön-İşleme

Giriş verisi, her bir makine öğrenimi görevine uygun miktar, yapı ve biçimde kusursuz bir şekilde sağlanmalıdır. Bu, özellikle sanal makine (VM) görevleri gibi bilgi işlem kaynaklarının verimli kullanımının kritik olduğu ortamlarda önemlidir ne yazık ki, gerçek dünya verileri gürültü (noise), eksik değerler (Missing Values- MVs), tutarsız ve gereksiz veriler ile örnek ve özellik boyutlarında devasa boyutlar gibi olumsuz faktörlerden büyük ölçüde etkilenir. Bu tür kusurlar, veri kalitesini düşürür ve doğrudan ML algoritmalarının performansını olumsuz etkiler. Bu nedenle, veri ön işleme herhangi bir ML projesinde önemli yer tutar. Veri temizlemenin temel amacı, ham veriyi daha tutarlı, güvenilir ve analiz için uygun bir biçime dönüştürmektir. Eksik değerlerin giderilmesi de bu sürecin önemli bir parçasıdır.

Eksik değer içeren değişkenlerle başa çıkmanın yaygın bir yaklaşımı, bu değerleri sezgisel veya istatistiksel yöntemlerle doldurmaktır. MV içeren değişkenleri sezgisel verilerle doldurmak çoğu durumda, uygun bir veri değerinin makul bir tahminini eklemek, boş bırakmaktan daha iyidir [9].

Bu çalışmada, veri temizleme sürecinde ilk olarak, kategorik veriyi, ML algoritmalarının işleyebileceği ve üzerinde analizleri daha etkili yapabileceği sayısal bir formata dönüştürme işlemi gerçekleştirilmiştir. Bu dönüşüm genellikle one-hot encoding gibi teknikler kullanılarak yapılır. Ardından, eksik değerler ele alınmıştır. Kategorik değişkenlerdeki eksik değerler en sık değerle (mod) doldurulurken, sayısal değişkenlerdeki eksik değerler ortalama (mean) ile doldurulmuştur. Bu seçim, kategorik verilerdeki baskın kategoriyi koruma ve sayısal verilerdeki dağılımı nispeten koruma amacını taşır. Bu tür veri ön işleme adımları, ML modellerinin daha doğru ve güvenilir sonuçlar üretmesine olanak tanır.

3 Kullanılan Sınıflandırma Yöntemleri

Bu projede, kronik böbrek hastalığını sınıflandırmak için çeşitli makine öğrenmesi algoritmaları karşılaştırmalı olarak kullanılmıştır. Kullanılan yöntemler; K-En Yakın Komşu (KNN), Naive Bayes, Karar Ağaçları, Destek Vektör Makineleri (SVM: Lineer, RBF, Polinomsal çekirdekli), Lojistik Regresyon, Random Forest ve Çok Katmanlı Algılayıcı (Multilayer Perceptron- MLP) algoritmalarıdır.

3.1.1 2.2.1 KNN

KNN algoritması, gözetimli (supervised) bir sınıflandırma algoritmasıdır. Algoritmanın çalışma prensibi, en yakın k komşuluktaki vektör temel alınarak sınıflandırmaya dayanmaktadır. ([10].

3.1.2 Native Bayes

Naive Bayes sınıflandırıcılar Bayes teoremine dayanarak üretilmiş olasılık tabanlı sınıflandırıcılardır. Sınıflandırıcı, üzerindeki her sınıf için olasılık değerlerini hesaplar ve sınıflandırılacak her veri için ihtimali en yüksek sınıfı bulmayı amaçlar [11].

3.1.3 Karar Ağaçları

Karar ağaçları ilk aşamada veriyi sınıflandırmak amacıyla ağaç yapısını üstten başlayarak oluşturan algoritmalardır. Ağaç yapısı en üsten başlamak koşulu ile kök, dal ve yapraklar olarak isimlendirilir. Her dal üsteki köke bağlı olmak koşuluyla dallar düğümlere bağlanır. Veride bulunan her bir öznitelik sınıflandırma sonrasında ağaçta bir düğüm noktasını temsil etmektedir. Tüm bu ağaç yapısı arasında kalan düğüm noktaları birer sınıflandırma kuralıdır ve her bir yaprak bir sınıf kabul edilir [12].

3.1.4 Lineer SVM:

Lineer Destek Vektör Makineleri, iki sınıf arasındaki ayrımı sağlamak için büyük marjlı bir sınıflandırıcı oluşturur. Hiper düzlem, $w \cdot x - b = 0$ ile tanımlanır.

3.1.5 Polinomal SVM (Polynomial Kernel):

Polinom çekirdek fonksiyonu, SVM'de örneklerin benzerliğini orijinal değişkenlerin polinomları üzerinden temsil eder. Derece d olan polinomlar için çekirdek, $K(x,y)=(xy+C)$ formülü ile tanımlanır

3.1.6 Radyal Tabanlı Fonksiyon (RBF) SVM:

RBF çekirdek, Euclidean mesafeye dayalı bir benzerlik fonksiyonudur ve $K(x,x')=\exp(-\gamma||x-x'||2)$ formülü ile tanımlanır [13].

3.1.7 Lojıstik Regresyon

Lojistik regresyon, ikili sonuçları analiz etmek için kullanılan, sürekli veya kategorik tahmin değişkenlerini içerebilen ve birden fazla tahmin değişkenine göre ayarlama yapabilen bir istatistiksel modeldir [14].

3.1.8 Random Forest

Rastgele orman modelleri, birden fazla karar ağacından oluşur; her ağaç, eğitim verisindeki değişkenlerin rastgele seçilen alt kümeleri kullanılarak oluşturulur [15].

3.1.9 MLP (Çok Katmanlı Algılayıcı)

MLP Yapay Sinir Ağları, insan beyninin bilgi işleme şeklini taklit eden ve makine öğrenmesi problemlerinde etkili çözümler sunan bir yapıdır. MLP, giriş, gizli ve çıktı katmanlarından oluşur ve ağırlıklar, biaslar ve aktivasyon fonksiyonları kullanarak öğrenir [16].

3.2 10 Katlı Çapraz Doğrulama

Verilerin eğitim ve test olarak ayrıştırılmasında 10-katlı çapraz doğrulama yöntemi kullanılmıştır. Bu yöntemde veriler öncelikle 10 ayrı gruba ayrılır ve bir grup test amaçlı kullanılırken geriye kalan diğer dokuz grup da eğitim amaçlı kullanılır. Tüm veriler hem test hem de eğitim amaçlı kullanılması için bu işlem 10 kez tekrarlanır ve gruplar değiştirilir [17].

Model doğrulama yöntemini kullanarak, istatistiksel analiz sonuçlarının bağımsız veri kümesi olarak basitleştirilebileceğini öngören yönteme çapraz doğrulama denilmektedir. Çapraz doğrulama, orijinal tahmin örneklerini sırasıyla eğitim ve test alt örneklerine ayırarak, test analizinin gerçekleştirilmesini içermektedir. Çapraz doğrulama yöntemi sınıflandırıcının değişkenliğini azaltmak için, eğitim ve test

alt örneklerinin farklı bölümlerini kullanarak birçok döngüde gerçekleştirilir. Döngüler tamamladıktan sonra doğrulama sonuçlarının ortalaması alınır [18].

3.3 Genetik Algoritma ile Feature Selection

Genetik algoritmalar, yüksek boyutlu ve çok sınıflı veri kümelerinde öznitelik seçimi için yaygın olarak kullanılan sezgisel yöntemlerdendir. GA, öznitelik değerlendirme ve sınıflandırma süreçlerini geri beslemeli bir yapı içinde birleştirerek hem öznitelik seçimini hem de sınıflayıcı tasarımını eş zamanlı olarak gerçekleştirebilir [19]. GA süreci; başlangıçta rastgele oluşturulan kromozomlardan (öznitelik kombinasyonları) oluşan bir popülasyonun oluşturulmasıyla başlar. Ardından her nesilde, uygunluk değeri yüksek bireyler seçilir ve genetik işlemler olan **çaprazlama (crossover)** ve **mutasyon (mutation)** uygulanarak yeni bireyler üretilir. Bu evrimsel süreç, maksimum nesil sayısına ulaşılana kadar devam eder [20].

Bu çalışmada, **kronik böbrek hastalığı** verisinde sınıflandırma performansını artırmak amacıyla öznitelik seçimi için **GA** tabanlı bir yöntem uygulanmıştır. Veri seti öncelikle eksik değerlerden arındırılmış, kategorik veriler etiketlenmiş ve sayısal veriler standartlaştırılmıştır. Ardından GA sürecinde her bireyin ikili dizilerle temsil edildiği 50 bireyden oluşan bir başlangıç popülasyonu ile başlatılmıştır. Bireyler, özniteliklerin seçim durumunu (0 veya 1) yansıtan kromozomlar olarak tanımlanmıştır. Çaprazlama (oran: 0.7) ve mutasyon (oran: 0.2) işlemleri sonucunda yeni nesiller üretilmiş ve toplam 40 nesil boyunca evrimsel süreç devam etmiştir.

Elde edilen en iyi birey ile seçilen öznitelik alt kümesi üzerinden veri yeniden oluşturulmuş ve ardından KNN, Naive Bayes, Karar Ağacı, Lojistik Regresyon, Random Forest, SVM (lineer, rbf, polinom) ve MLP gibi dokuz farklı sınıflayıcı model eğitimi yapılmıştır. Modellerin başarıları **doğruluk (accuracy)**, **kesinlik (precision)**, **duyarlılık (recall)**, **özgüllük (specificity)**, **F1 skoru** ve **Matthews korelasyon katsayısı (MCC)** gibi çeşitli performans metrikleriyle değerlendirilmiştir. Bu yöntem sayesinde gereksiz özniteliklerin etkisi azaltılarak model genelleme başarısı artırılmış, aynı zamanda daha az karmaşık modellerle benzer veya daha iyi performans elde edilmiştir.

3.4 PCA ile Featrue Reduction

PCA, yüksek boyutlu verileri daha düşük boyutlu bir alt uzaya projekte ederek en yüksek varyansa sahip yönleri koruyan doğrusal bir dönüşümdür. Bu sayede öznitelik sayısı azaltılırken, verinin temel yapısı korunur. PCA yöntemi, veri kümesinin kovaryans matrisinin özvektörlerini ve özdeğerlerini hesaplayarak gerçekleştirilir. Bu dönüşüm, özellikle verideki ikinci dereceden korelasyonları kaldırarak, daha bağımsız ve anlamlı bileşenlerin elde edilmesini sağlar [21].

Bu çalışmada PCA yöntemi, veride daha kompakt ve anlamlı temsiller elde etmek amacıyla uygulanmıştır. Öncelikle tüm bileşenler üzerinden PCA analizi gerçekleştirilmiş ve her bileşenin açıkladığı varyans oranı hesaplanmıştır. Kümülatif varyans incelenerek, toplam varyansın %95'inden fazlasını temsil eden minimum bileşen sayısı belirlenmiştir. Bu analiz sonucunda, 24 öznitelikten oluşan veri seti 20 bileşene indirgenmiş ve böylece toplam varyansın %95.90'ı korunmuştur. Sonuç olarak, PCA uygulaması ile bilgi kaybı minimize edilerek veri boyutu azaltılmış ve sonraki sınıflandırma modelleri için daha sade ve etkili bir veri temsili elde edilmiştir.

4 Bulgular ve Tartışma

Bu projenin test ortamı, kronik böbrek hastalığı tanısı için geliştirilen makine öğrenimi modellerinin performansını ve güvenilirliğini değerlendirmek üzere tasarlanmıştır. Öneri sistemlerini 9 tane sınıflanıdırıcı (KNN, Naive Bayes, Karar Ağaçları, SVM, Lojistik Regresyon, Rastgele Orman, MLP) üzerinden 10 kat çaprazlama tekniği, GA ile feature selection, PCA ile feature reduction yöntmeleri ile test ettik ve performans sonuçları elde ettik.

4.1 Karmaşıklık Matrisi

İkili bir sınıflandırıcı için karmaşıklık matrisi (Şekil 4-1). Gerçek değerler Doğru (1) ve Yanlış (0) olarak işaretlenmiş, tahmin edilen değerler ise Pozitif (1) ve Negatif (0) olarak belirtilmiştir. Sınıflandırma modellerinin olanaklarına ilişkin tahminler, **karmaşıklık matrisinde** bulunan **TP, TN, FP, FN** ifadelerinden türetilir.

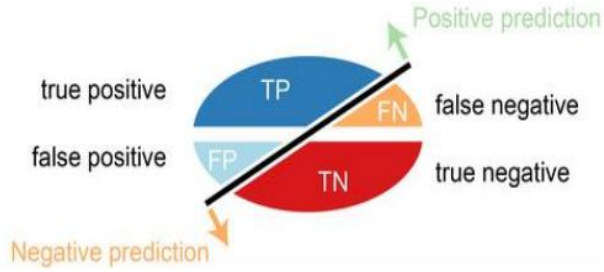
Class designation		Actual class	
		True (1)	False (0)
Predicted class	Positive (1)	TP	FP
	Negative (0)	FN	TN

Şekil 4-1: Karmaşıklık matrisi ve temel bileşenleri

Karmaşıklık matrisindeki şekilde de göründüğü gibi dört temel bileşen şunlardır:

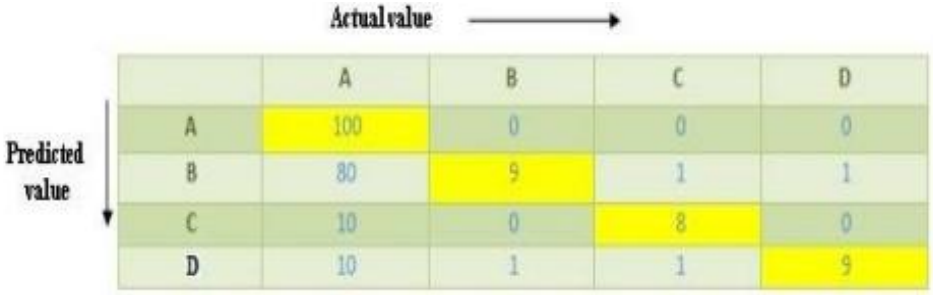
- **TP (True Positive- Gerçek Pozitif):** Karmaşıklık matrisindeki veri noktası, pozitif bir sonuç tahmin edildiğinde ve gerçekleşen durumun da aynı (pozitif) olduğunda Gerçek Pozitiftir.
- **FP (False Positive- Yanlış Pozitif):** Karmaşıklık matrisindeki veri noktası, pozitif bir sonuç tahmin edildiğinde ve gerçekleşen durumun negatif bir sonuç olduğunda yanlış pozitiftir.
- **FN (False Negative- Yanlış Negatif):** Karmaşıklık matrisindeki veri noktası, negatif bir sonuç tahmin edildiğinde ve gerçekleşen durumun pozitif bir sonuç olduğunda yanlış negatiftir.

- **TN (True Negative- Gerçek Negatif):** Karmaşıklık matrisindeki veri noktası, negatif bir sonuç tahmin edildiğinde ve gerçekleşen durumun da aynı (negatif) olduğunda Gerçek Negatiftir. Şekil 4-2'de gösterilmiştir.



Şekil 4-2: Sınıflandırıcıların 4 sonucu

Dört sınıflı sınıflandırma için karmaşıklık matrisi (Şekil 4-3'te). Dört sınıflı sınıflandırma, örnekleri dört sınıfa ayırma problemidir. Dört sınıf durumu: sınıf A, sınıf B, sınıf C ve sınıf D.

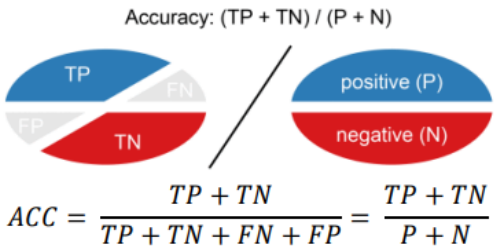


Şekil 4-3 Örnek karmaşıklık matrisi

4.2 Performans Metrikleri [22]

- Doğruluk (Accuracy)

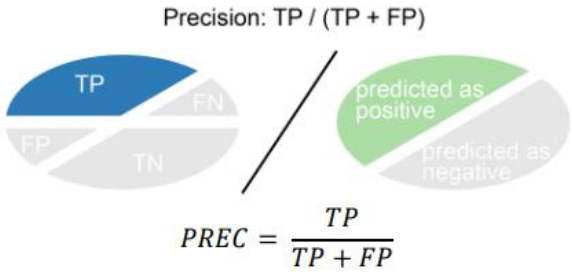
Doğruluk, iki doğru tahminin (TP + TN) toplamının veri setlerinin toplam sayısına (P + N) bölünmesiyle hesaplanır. En iyi doğruluk 1.0 ve en kötüsü 0.00'dır (Şekil4-4).



Şekil 4-4: Doğruluk formülü

- Kesinlik (Precision)

Kesinlik, doğru pozitif tahminlerin (TP) sayısının, toplam pozitif tahminlere (TP + FP) bölünmesiyle hesaplanır (Şekil 4-5). En iyi doğruluk 1.0 ve en kötüsü 0.0'dır.

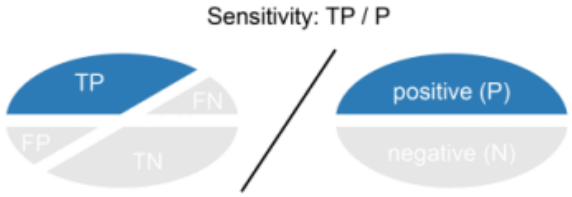


$$PREC = \frac{TP}{TP + FP}$$

Şekil 4-5: Kesinlik formülü

- Duyarlılık (Recall)

Gerçek Pozitif Oranı (Hassasiyet veya Duyarlılık), doğru pozitif tahminlerin sayısının toplam pozitiflerin sayısına bölünmesiyle hesaplanır. Aynı zamanda Hassasiyet veya Duyarlılık (REC) olarak da adlandırılır. En iyi Recall Oranı 1.0 ve en kötüsü 0.0'dır.

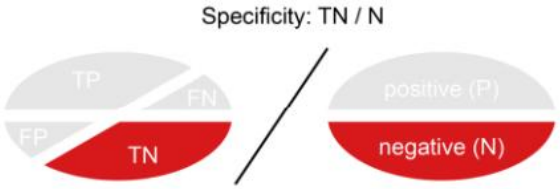


$$SN = \frac{TP}{TP + FN} = \frac{TP}{P}$$

Şekil 4-6: Hassasiyet hesaplaması formülü

- Özgüllük (Specificity)

Gerçek Negatif Oranı -TNR (Özgüllük)- doğru negatif tahminlerin (TN) sayısının toplam negatiflerin (N) sayısına bölünmesiyle hesaplanır. En iyi özgüllük 1.0 ve en kötüsü 0.0'dır. (Şekil 4-7)



$$SP = \frac{TN}{TN+FP} = \frac{TN}{N} \qquad SP = 1 - FPRate$$

Şekil 4-7: Specificity hesaplama formülü

- F1 skoru (F1-score)

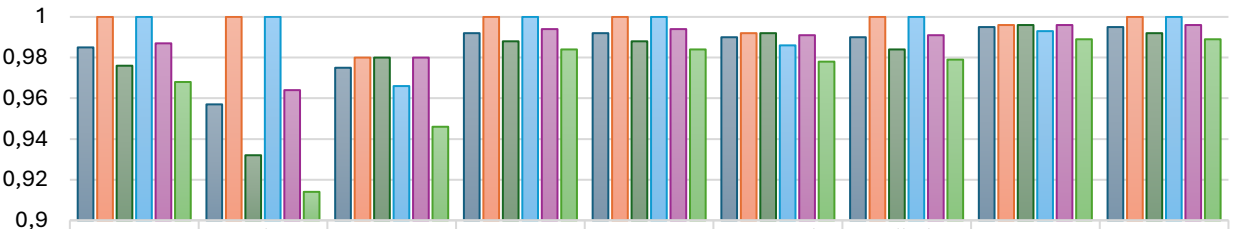
F-Ölçüsü veya F-skoru, testin doğruluğunun bir ölçüsüdür. Kesinlik ve duyarlılığa dayalı olarak şu formülle hesaplanır:

Score = (2 * precision * recall) / (precision + recall)

- MCC

Matthews Korelasyon Katsayısı (MCC) - tahmin edilen sınıflar ile temel gerçek arasındaki korelasyondur. Karmaşıklık matrisindeki değerlere göre hesaplanır. MCC = (TP*TN - FP*FN) / karekök((TP + FP)(TP + FN)(TN + FP)(TN + FN)) MCC genellikle dengeli bir ölçü olarak kabul edilir ve sınıfların boyutları çok farklı olsa bile kullanılabilir.

4.3 Modelleme Sonuçları

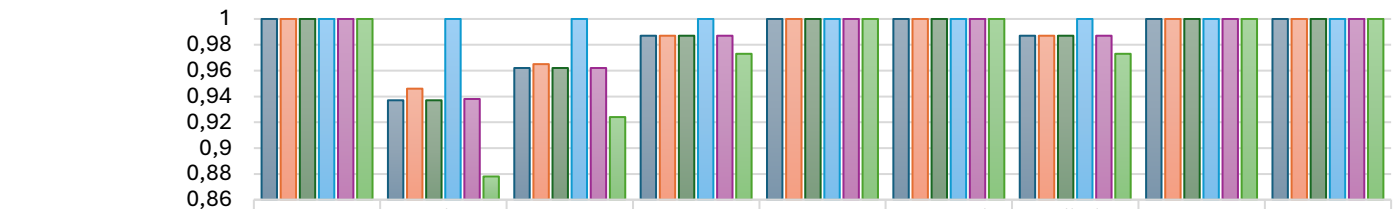


	KNN	Naive Bayes	Karar Ağacı	Lineer SVM	RBF SVM	Polynomial SVM	Lojistik Regresyon	Random Forest	MLP
Doğruluk (Accuracy)	0,985	0,957	0,975	0,992	0,992	0,99	0,99	0,995	0,995
Kesinlik (Precision)	1	1	0,98	1	1	0,992	1	0,996	1
Duyarlılık (Recall)	0,976	0,932	0,98	0,988	0,988	0,992	0,984	0,996	0,992
Özgüllük (Specificity)	1	1	0,966	1	1	0,986	1	0,993	1
F1 skoru (F1-score)	0,987	0,964	0,98	0,994	0,994	0,991	0,991	0,996	0,996
MCC	0,968	0,914	0,946	0,984	0,984	0,978	0,979	0,989	0,989

Grafik 1: 10 Katlı çapraz doğrulama ile yapılan model sonuçları

10 Katlı Çapraz Doğrulama grafik 1'de temel performans seviyesini göstermektedir. Herhangi bir özellik seçimi veya boyut indirgeme uygulanmadan, 9 algoritmanın 10 katlı çapraz doğrulama ile elde edilen sonuçları şu şekildedir:.

- En yüksek performansı gösteren algoritmalar: Random Forest ve MLP'dir. Bu algoritmalar, Doğruluk, F1 Skoru ve MCC gibi metriklerde 0,99'un üzerinde değerler elde etmiştir.
- En düşük performansı gösteren algoritma: Naive Bayes'dir. Bu algoritma, diğerlerine kıyasla daha düşük Doğruluk (0,957) ve F1 Skoru (0,964) elde etmiştir.
- Diğer algoritmalar da (KNN, Karar Ağacı, Lineer SVM, RBF SVM, Polynomial SVM, Lojistik Regresyon) genel olarak yüksek performans göstermiştir.

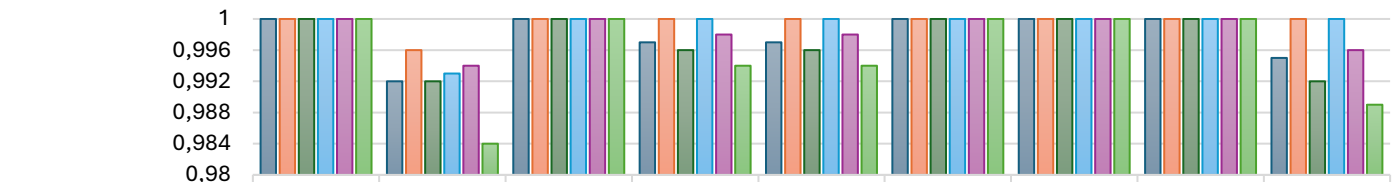


	KNN	Naive Bayes	Karar Ağacı	Lineer SVM	RBF SVM	Polynomial SVM	Lojistik Regresyon	Random Forest	MLP
Doğruluk (Accuracy)	1	0,937	0,962	0,987	1	1	0,987	1	1
Kesinlik (Precision)	1	0,946	0,965	0,987	1	1	0,987	1	1
Duyarlılık (Recall)	1	0,937	0,962	0,987	1	1	0,987	1	1
Özgüllük (Specificity)	1	1	1	1	1	1	1	1	1
F1 skoru (F1-score)	1	0,938	0,962	0,987	1	1	0,987	1	1
MCC	1	0,878	0,924	0,973	1	1	0,973	1	1

Grafik 2: GA ile Featur Selection yapılması sonucu elde edilen sonuçlar

GA ile Öznitelik Seçimi grafik 2'de GA kullanılarak yapılan öznitelik seçiminden sonraki sonuçları göstermektedir. GA, en iyi özellik alt kümesini seçerek modelin performansını artırmayı amaçlar.

- GA'nın etkisi: GA, bazı algoritmaların performansını artırmış, bazılarının performansını ise değiştirmemiştir. Random Forest ve MLP zaten diğerlerinden daha yüksek performans gösterdiği için diğer algotimaların GA'dan etkilenme oranına göre bu iki algoritma GA'dan etkilnememiş gibi hareket etmiştir ancak bu iki algoritmada da küçük de olsa iyileşme gözlenmlenmektedir.
- En yüksek performansı gösteren algoritmalar: KNN, RBF SVM, Polynomial SVM, Random Forest ve MLP bu tabloda da en iyi performansı göstermektedir.
- Naive Bayes'in performansı GA ile artmış ancak hala diğerlerine göre düşüktür.



	KNN	Naive Bayes	Karar Ağacı	Lineer SVM	RBF SVM	Polynomial SVM	Lojistik Regresyon	Random Forest	MLP
Doğruluk (Accuracy)	1	0,992	1	0,997	0,997	1	1	1	0,995
Kesinlik (Precision)	1	0,996	1	1	1	1	1	1	1
Duyarlılık (Recall)	1	0,992	1	0,996	0,996	1	1	1	0,992
Özgüllük (Specificity)	1	0,993	1	1	1	1	1	1	1
F1 skoru (F1-score)	1	0,994	1	0,998	0,998	1	1	1	0,996
MCC	1	0,984	1	0,994	0,994	1	1	1	0,989

Grafik 3: PCA ile Feature Reduction yapılması ile elde edilen sonuçlar

PCA ile Boyut İndirgeme grafik 3'te PCA kullanılarak sonuçları göstermektedir. PCA, verideki önemli bilgileri koruyarak özellikleri birbirlerinden bağımsız hale getirir ve bunun sonuçlarından biri olarak özellik sayısını azaltır.

- PCA'nın etkisi, algoritmaların performansını genel olarak yüksek tutmuştur.
- En yüksek performansı gösteren algoritmalar: KNN, Karar Ağacı, Polynomial SVM, lojistik Regresyon ve Random Forest
- Naive Bayes diğer algoritmalar kadar iyi performans göstermese de PCA ile en iyi performansına sahiptir.

Genel Sonuç

- En iyi algoritmalar: Random Forest ve MLP, her üç senaryoda da (10 katlı çapraz doğrulama, GA ile öznitelik seçimi, PCA ile boyut indirgeme) en iyi performansı göstermiştir. Bu algoritmalar, KBH teşhisi için oldukça etkili olabilir.
- GA ve PCA'nın etkisi: GA ve PCA, bazı algoritmaların performansını artırsa da Random Forest ve MLP gibi zaten yüksek performans gösteren algoritmalar üzerinde önemli bir etkisi olmamıştır.
- Naive Bayes en düşük performansı göstermiştir.

5 Sonuçlar

Bu çalışma, KBH tanısı için farklı makine öğrenmesi algoritmalarının performansını karşılaştırmıştır. University of California Irvine (UCI) makine öğrenmesi deposundan elde edilen açık kaynaklı bir veri seti kullanılarak, 9 farklı algoritma 10 katlı çapraz doğrulama, genetik algoritmalar (GA) ile özellik seçimi ve PCA ile boyut indirgeme yöntemleri altında değerlendirilmiştir.

Çalışmanın temelinde e**n iyi performansı** Random Forest ve MLP, tüm senaryolarda (10 katlı çapraz doğrulama, GA ve PCA) en yüksek doğruluk, kesinlik, duyarlılık, F1 skoru ve MCC değerlerini elde etmiş olup KBH tanısında yüksek doğruluk ve güvenilirlik sağladığını göstermektedir.

Bu çalışmada, KBH tanısında makine öğrenmesi algoritmalarının potansiyelini ortaya koymaya çalışılmıştır. Elde edilen sonuçlara göre, özellikle Random Forest ve MLP algoritmalarının, KBH'nin doğru teşhisi için kullanılabilecek etkili algortimalar olduğu saptanmıştır. Bu bulgular, gelecekteki araştırmalar ve uygulamalar için bir temel oluşturabilecek potansiyele sahiptir.

İleride daha kapsamlı veri setlerinde farklı demografik gruplardan ve coğrafi bölgelerden elde edilen daha geniş ve çeşitli veri setleri kullanılarak, modellerin genelleme yeteneği artırılabilir ve farklı popülasyonlardaki KBH vakalarını daha doğru bir şekilde tespit edebilir. Özellikle MLP'nin başarısı göz önünde bulundurularak, KBH tanısında derin öğrenme modellerinin (örneğin, evrişimsel sinir ağları veya tekrarlayan sinir ağları) potansiyeli araştırılabilir. Bu modeller, daha karmaşık ilişkileri öğrenebilir ve daha yüksek doğruluk sağlayabilir.

6 Kaynakça

[1] V. Jha, G. Garcia-Garcia, K. Iseki, Z. Li, S. Naicker, B. Plattner, R. Saran, A. Y.-M. Wang ve C.-W. Yang, «Chronic kidney disease: global dimension and perspectives,» %1 içinde *Global Kidney Disease 3*, 2013.

[2] J. Qin, L. Chen, Y. Liu, C. Liu, C. Feng ve B. Chen, «A Machine Learning Methodology for Diagnosing Chronic Kidney Disease,» 30 Aralık 2020. [Çevrimiçi]. Available: https://ieeexplore.ieee.org/document/8945312.

[3] A. Subasi, E. Alickovic ve J. Kevric, «Diagnosis of Chronic Kidney Disease by Using Random Forest,» %1 içinde *IFMBE Proceedings*, 2017.

[4] L. Zhang, F. Wang, L. Wang, W. Wang, B. Liu ve J. Liu, «Prevalence of chronic kidney disease in China: a cross-sectional survey,» 2012.

[5] A. Singh, G. Nadkarni, O. Gottesman, S. B. Ellis, E. p. Bottinger ve J. V. Guttag, «Incorporating temporal EHR data in predictive models for risk stratification of renal function deterioration,» 15 Kasım 2014. [Çevrimiçi]. Available: https://pubmed.ncbi.nlm.nih.gov/25460205/.

[6] A. M. Cueto-Manzano, L. Cortes-Sanabria, H. R. Martinez-Ramirez, E. Rojas-Compas, B. Gomez-Navarro ve M. Castillero-Manzano, «Prevalence of chronic kidney disease in an adult population,» 1 Haziran 2014. [Çevrimiçi]. Available: https://pubmed.ncbi.nlm.nih.gov/24992221/.

[7] H. Polat, A. Çetin ve H. D. Mehr, *Journal of Medical Systems,* Springer, 2017.

[8] M. Iqbal, «Chronic Kidney Disease,» 2017. [Çevrimiçi]. Available: https://www.kaggle.com/datasets/mansoordaku/ckdisease.

[9] S. Garcia, J. Luengo ve F. Herrera, Data Preprocessing in Data Mining, Granada,Spain: Springer, 2015.

[10] R. O. Duda, P. E. Hart ve D. G. Stork, Pattern Classification 2nd Edition, New York: Wiley Interscience, 2001.

[11] B. Baharudin, A. Khan, L. H. Lee ve K. Khan, «A Review of Machine Learning Algorithms for Text-Documents Classification,» 1 Şubat 2010. [Çevrimiçi]. Available: https://www.jait.us/index.php?m=content&c=index&a=show&catid=160&id=859.

[12] S. B. Kotsiantis, «Decision trees: a recent overview,» %1 içinde *Artificial Intelligence Review*, Springer, 2013, pp. 261-283.

[13] S. Ghosh, A. Dasgupta ve A. Swetapadma, «A Study on Support Vector Machine based Linear and Non-Linear Pattern Classification,» %1 içinde *2019 International Conference on Intelligent Sustainable Systems (ICISS)*, Palladam, India, 2019.

[14] M. P. LaValley, «Circulation,» 6 Mayıs 2008. [Çevrimiçi]. Available: https://www.ahajournals.org/doi/full/10.1161/CIRCULATIONAHA.106.682658.

[15] S. Pichhadze ve B. Kaplan, «Seeing the Forest for the Trees: Random Forest Models for Predicting Survival in Kidney Transplant Recipients,» 22 Mayıs 2020. [Çevrimiçi]. Available: https://journals.lww.com/transplantjournal/FullText/2020/05000/Seeing_the_Forest_for_the_Trees__Random_Forest.8.aspx.

[16] Baristuzemenn, «MLP (Çok Katmanlı Algılayıcı) Yapay Sinir Ağları,» 19 Aralık 2023. [Çevrimiçi]. Available: https://medium.com/@baristuzemenn/mlp-%C3%A7ok-katmanl%C4%B1-alg%C4%B1lay%C4%B1c%C4%B1-yapay-sinir-a%C4%9Flar%C4%B1-a8c9c68748f6.

[17] E. Aydemir, «Ders Geçme Notlarının Veri Madenciliği Yöntemleriyle Tahmin Edilmesi,» 2019.

[18] U. Desai, «Automated detection of cardiac health condition using linear techniques,» %1 içinde *2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, 2017.

[19] M. Pei, E. Goodman ve W. Punch, «Feature Extraction Using Genetic Algorithms,» Eylül 1998. [Çevrimiçi]. Available: 2779950_Feature_Extraction_Using_Genetic_Algorithms.

[20] K. A. Shastry ve H. A. Sanjau, «Knowledge-Based Systems: A modified genetic algorithm and weighted principal component analysis based feature selection and extraction strategy in agriculture,» 28 Kasım 2021. [Çevrimiçi]. Available: https://www.sciencedirect.com/science/article/pii/S095070512100722X.

[21] J. Qiu, H. Wang, J. Lu, B. Zhang ve K. L. Du, «Neural Network Implementations for PCA and Its Extensions,» %1 içinde *International Scholarly Research ISRN Artificial Intelligence*, 2012.

[22] Z. Vujovic, «Classification Model Evaluation Metrics,» %1 içinde *International Journal of Advanced Computer Science and Applications Volume 12(Issue 6)*, 2021.

[23] L. Lei ve P. Liu, «Missing Data Treatment Methods and NBI Model,» 11 Kasım 2006. [Çevrimiçi]. Available: https://ieeexplore.ieee.org/document/4021513.

[24] L. Breiman, «Random Forests,» %1 içinde *Machine Learning*, Berkeley, Springer, 2001, pp. 1-33.

[25] S. Özdemir, «Random Forest Yöntemi kullanılarak potansiyel dağılım modellemesi ve haritalaması: Yukarıgökdere Yöresi örneği,» 31 Mart 2018. [Çevrimiçi]. Available: https://doi.org/10.18182/tjf.342504.