ÖZELLİK SEÇİM YÖNTEMLERİ KULLANILARAK SINIFLANDIRMA ALGORİTMALARININ PERFORMANSLARININ KARŞILAŞTIRILMASI

YÜKSEK LİSANS TEZİ

Mustafa DEMİR

Danışman
Prof. Dr. İbrahim KILIÇ

İSTATİSTİK ANABİLİM DALI

Temmuz 2021

AFYON KOCATEPE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

ÖZELLİK SEÇİM YÖNTEMLERİ KULLANILARAK SINIFLANDIRMA ALGORİTMALARININ PERFORMANSLARININ KARŞILAŞTIRILMASI

Mustafa DEMİR

Danışman
Prof. Dr. İbrahim KILIÇ

İSTATİSTİK ANABİLİM DALI

Temmuz 2021

TEZ ONAY SAYFASI

Mustafa DEMİR tarafından hazırlanan "Özellik Seçim Yöntemleri Kullanılarak Sınıflandırma Algoritmalarının Performanslarının Karşılaştırılması" adlı tez çalışması lisansüstü eğitim ve öğretim yönetmeliğinin ilgili maddeleri uyarınca 05/07/2021 tarihinde aşağıdaki jüri tarafından **oy birliği** ile Afyon Kocatepe Üniversitesi Fen Bilimleri Enstitüsü **İstatistik Anabilim Dalı'nda YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Danışman : Prof. Dr. İbrahim KILIÇ **İmza**

Başkan : Doç. Dr. Sinan SARAÇLI
Afyon Kocatepe Üniversitesi, Fen-Edebiyat Fakültesi

Üye : Prof. Dr. İbrahim KILIÇ
Afyon Kocatepe Üniversitesi, Veteriner Fakültesi

Üye : Doç. Dr. Cengiz GAZELOĞLU
Süleyman Demirel Üniversitesi, Fen-Edebiyat Fakültesi

Afyon Kocatepe Üniversitesi
Fen Bilimleri Enstitüsü Yönetim Kurulu'nun
...../...../..... tarih ve
.....sayılı kararıyla onaylanmıştır.

.....
Prof. Dr. İbrahim EROL
Enstitü Müdürü

BİLİMSEL ETİK BİLDİRİM SAYFASI
Afyon Kocatepe Üniversitesi

Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- Tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- Görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- Başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- Atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- Kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- Ve bu tezin herhangi bir bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

05/07/2021

Mustafa DEMİR

ÖZET

Yüksek Lisans Tezi

Özellik Seçim Yöntemleri Kullanılarak Sınıflandırma Algoritmalarının Performanslarının Karşılaştırılması

Mustafa DEMİR
Afyon Kocatepe Üniversitesi
Fen Bilimleri Enstitüsü
İstatistik Anabilim Dalı
Danışman: Prof. Dr. İbrahim KILIÇ

Bu çalışmanın amacı, istatistik biliminde büyük önem teşkil eden sınıflandırma yöntemleri için ilgili veri setindeki değişkenlerin farklı özellik seçim yöntemleri ile belirlenmesidir. Özellik seçim yöntemleri k adet değişken seti içerisinden veri yapısına en uygun daha az sayıda değişkenin belirlenmesinde kullanılan ve sağladığı avantajlar bakımından da son yıllarda popülerliği artan istatistiksel yöntemler bütünüdür. Özellik seçim yöntemleri içerisinde kullanılan farklı teknikler, farklı sayıda ve farklı değişkenlerin seçilmesine sebep olabilmektedir. Bu çalışmada ilk olarak farklı tekniklere yardımıyla yapılan özellik seçimi sonucunda elde edilen yeni veri setleri oluşturulmuştur. Daha sonra oluşturulan bu veri setleri farklı makine öğrenme teknikleri ile analiz edilerek ilgili veri seti için yapılan karşılaştırmalar sonucunda en iyi makine öğrenme tekniği belirlenmiştir. Çalışmada kronik böbrek hastalığı veri seti kullanılarak Weka paket programı yardımı ile ilgili analizler gerçekleştirilmiştir. Analiz sonuçlarına göre korelasyon tabanlı özellik seçim yöntemi uygulandığında en iyi doğru sınıflandırma oranı %99.75 ile rassal orman ve çok katmanlı algılayıcı, filtre özellik seçim yöntemi uygulandığında %99.75 ile k-en yakın komşu, tutarlılık özellik seçim yöntemi uygulandığında %98.75 ile rassal orman en yüksek doğru sınıflandırma oranına sahiptir. Tutarlılık özellik seçim yöntemi uygulandığında %89 ile destek vektör makineleri(RTF Kernel) en düşük doğru sınıflandırma oranını vermiştir. Bu çalışmadan elde edilen bulgular incelendiğinde aynı veri seti kullanılarak yapılan daha önceki çalışmalara

nazaran daha yüksek doğru sınıflama oranları elde edilmiştir. Çalışmadan elde edilen diğer bulgu ve sonuçlar ilgili çizelge ve şekillerde sunulmuştur.

2021, xi + 65 sayfa

Anahtar Kelimeler: Sınıflandırma yöntemleri, Özellik seçim yöntemi, Makine öğrenmesi, Performans, Kronik böbrek hastalığı, Weka.

ABSTRACT

M.Sc. Thesis

Comparison of The Performances of Classification Algorithms Using Feature Selection Methods

Mustafa DEMİR

Afyon Kocatepe University

Graduate School of Natural and Applied Sciences

Department of Statistics

Supervisor: Prof. İbrahim KILIÇ

The purpose of this study is to determine the variables in the relevant data set with different feature selection methods for classification methods, which are of great importance in statistics**.** Feature selection methods are a set of statistical methods, which are used to determine less number of variables that are most suitable for the data structure among k variable sets and have become popular in recent years in terms of their advantages. Different techniques used in feature selection methods may cause the selection of different numbers and different variables. In this study, firstly, new data sets obtained as a result of feature selection made with the help of different techniques were created. Afterwards, these data sets created were analyzed with different machine learning techniques and the best machine learning technique was determined as a result of the comparisons made for the relevant data set. In this study by using the cronic kidney data set, analyzes were carried out with the help of Weka software. According to the results of the analysis, the best correct classification rate is random forest and multilayer perceptron with 99.75% when the correlation-based feature selection method is applied, the k-nearest neighbor with 99.75% when the filter feature selection method is applied, and the random forest with 98.75% when the consistency feature selection method is applied. It has the correct classification rate. When the consistency feature selection method was applied, support vector machines (RBF Kernel) gave the lowest correct classification rate with 89%.

The results of this study indicate that compared with the earlier studies using the same data set, the accuracy ratios of this study are much greater than the others. The other results obtained from this study are given in related tables and figures.

2021, xi + 65 pages

Key Words: Clustering methods, Future selection methods, Machine learning, Performance, Chronic kidney disease, Weka.

TEŞEKKÜR

Bu tezin her aşamasında maddi manevi desteğini esirgemeyen değerli büyüğüm ve danışmanım Sayın Prof. Dr. İbrahim KILIÇ'a,

Zorlu geçen bu süreçte, her türlü desteğiyle yanımda olan eksikliğini hissettirmeyen, tüm bilgi birikimini aktarmada bir an olsun tereddüt etmeyen, ahlaki değerleriyle örnek edindiğim, yüksek lisans süresince yanında bulunmaktan ve öğrencisi olmaktan dolayı onur duyduğum saygıdeğer hocam Doç. Dr. Sinan Saraçlı'ya,

Bu tez konusunun belirlenmesi ve sonuca ulaşması için bana yol gösteren değerli vaktini ayıran tüm desteğiyle yanımda olan Sayın Doç. Dr. Cengiz GAZELOĞLU'na,

Yine bu süreçte bana destek olan Sayın Hocam Öğretim Görevlisi Uğur Erdem ÜRER ve arkadaşlarım Nefise FERMANCI ile Şevkiye BABACAN'a,

Tezimin başlangıcından bitimine kadar bana inanan, yanımda olduğunu hissettiren, gösterdiği anlayış ve destekle, sevgideğer arkadaşım Feride SÖZER'e,

En içten teşekkürlerimi sunarım.

Her zaman yanımda olduklarını hissettiğim bugünlere gelmemde üzerimde büyük emekleri olan, başta abim Mehmet DEMİR ve ablam Havva DEMİR olmak üzere aileme şükranlarımla.

Mustafa DEMİR

AFYONKARAHİSAR 2021

İÇİNDEKİLER DİZİNİ

Sayfa

SİMGELER ve KISALTMALAR DİZİNİ

Simgeler

χ^2	Ki-kare
$\phi(.)$	Dönüşüm fonksiyonu
$\varphi_{(i)}$	Aktivasyon fonksiyonu
P_0	Kabul edilen oran
P_c	Beklenen oran
κ	Kappa değeri
G^T	G matrisinin transpozu
K (x, x_i)	Kernel fonksiyonu
$d(x,y)$	Öklid uzaklığı
y_i^h	h. katmandan önceki i. nöron
w_{ji}^h	Nöronlar arası ağırlık
b_0	Eşik değer
Y_n	Normalleştirilmiş çıktılar
$Y_i(x)$	i. radyal temelli fonksiyon
f_0	Çıktı katmanı transfer fonksiyonu
f_n	Gizli katman transfer fonksiyonu
b_{hk}	k. gizli katmanın bias terimleri
x_{ni}	Normalize edilmiş girdi vektörü
P(H|X)	Bayes teoremi
x^+	En iyi özellik
x^-	En kötü özellik
N	Anakütle hacmi
Y_0	Boş küme
A_{ij}	Gözlenen değer
E_{ij}	Beklenen değer
S	Özellik alt kümesi
ξ	Sınıf niteliği
Ω	Sayılabilir anakütle özelliği

Kısaltmalar

AGYS	Ardışık geri yönde seçim
AGYKS	Ardışık geri yönde kayan seçim
AİYS	Ardışık ileri yönde seçim
AİYKS	Ardışık ileri yönde kayan seçim
ÇKA	Çok katmanlı algılayıcı
DP	Doğru pozitif
DVM	Destek vektör makineleri
EAA	Eğri altındaki alan
KBH	Kronik böbrek hastalığı
k-NN	k-En yakın komşular
NB	Naive bayes
ÖS	Özellik seçimi
RTF	Radyal temelli fonksiyon

TÖ	Tutarlılık ölçütü
YP	Yanlış pozitif
YSA	Yapay Sinir Ağları

ŞEKİLLER DİZİNİ

Sayfa

ÇİZELGELER DİZİNİ

Sayfa

1. GİRİŞ

Gerçek hayatta karşılaştığımız problemlerde çoğunlukla birbiriyle ilişkili özellikler hakkında yeterli bilgi bulunmaz. Bu nedenle, ilgilenilen problemi daha iyi temsil edebilmek için pek çok aday özellik belirlenir. Ancak bu durum da gereksiz özelliklerin seçimi ile sonuçlanır. Gereksiz özellikler bağımlı değişkenle direkt olarak ilişkili olmayan özelliklerdir ancak öğrenme sürecini etkiler ve amaca herhangi bir katkı sağlamaz. Günümüzde pek çok sınıflandırma probleminde kullanılan veri büyük boyutlu olduğundan istenmeyen özellikler çıkarılmadan iyi bir sınıflandırıcı elde etmek zordur. Gereksiz ya da ilgisiz özelliklerin sayısının düşürülmesi öğrenme algoritmasının hem çalışma süresini kısaltır hem de genelleştirme başarısının daha yüksek olmasını sağlar. Böylelikle gerçek hayat sınıflandırma problemine daha iyi bir yaklaşım ve bakış açısı geliştirilmiş olur (Ay 2019).

Özellik Seçimi (ÖS) makine öğrenmesinde yaygın olarak kullanılan bir yöntem olup literatürde alt küme seçimi olarak bilinmektedir. ÖS işleminde, veri kümesinden elde edilen özellik alt kümesi, öğrenme algoritması için seçilir. Çözüm uzayı için en yüksek doğruluk oranına sahip olan en küçük boyutlu veri kümesinden oluşan küme en iyi alt küme olarak kabul edilir. Veri kümesindeki geriye kalan önemsiz özellikler yok sayılır. Bu aşama, önemli bir veri ön işleme aşamasıdır. ÖS'nin temel hedefi, orijinal özelliklerin hepsini kullanmadan en yüksek oranda veri bütünlüğünü sağlamaktır. Bununla birlikte minimum özellik alt kümesini bulmaktır. Birçok gerçek dünya probleminde, gereksiz, yanıltıcı veya gürültülü verinin çokluğundan dolayı ÖS bir zorunluluk olarak kabul edilmektedir. ÖS sonuçlarında en ideal çözümü bulmak için, tüm özellik alt kümelerinin test edilmesi gerekmektedir (Koç 2016).

Forman (2003)'a göre ös'nin temel amacı, performansı etkilemeden orijinal veri kümesini temsil edebilecek en iyi alt kümeyi seçme işlemi olarak tanımlanmaktadır. ÖS (nitelik seçimi veya değişken seçimi), kullanılacak algoritmaya uygun özellikleri değerlendirerek veri kümesindeki n adet özellik içerisinden en iyi k adedi seçme işlemi olarak tanımlanmaktadır (Karakaş 2020).

Teknolojide meydana gelen değişim, veri madenciliği süreçlerinde yüksek işlem kapasiteli bilişim sistemlerinin bütünleşik kullanımını sağlayarak veri madenciliği süreçlerini makine öğrenmesi alanında ele alma imkânı sağlamıştır (Beyazıt 2019).

Sınıflandırma, veri madenciliğinin en tanınmış işidir. Girdilerin çeşitli özelliklere göre bir sınıflayıcı (model) tarafından sınıflara atanması sürecidir. Eldeki nesnelerin bir sınıfa atanıp atanmayacağının ya da sınıflardan hangisine atanacağının belirlenmesidir. Başka bir ifade ile nesneler veya durumlar için uygun sınıfın tahmin edilmesidir. Sınıflama girdileri, her biri bir sınıf etiketi ile etiketlenecek gözlem veya örneklerden oluşan bir eğitim kümesidir. Çıktı ise modelin her bir gözlemlenen özelliklerine dayalı olarak atadığı sınıf etiketidir (Emel ve Taşkın 2005).

Kronik Böbrek Hastalığı (KBH), böbreğin sıvı çözünen dengesinde ve metabolik-endokrin fonksiyonlarda glomerüler filtrasyon değerinin azalması sonucu kronik ve ilerleyici bozulma olarak tanımlanır. Son dönemlerde böbrek yetmezliği, glomerüler filtrasyon değeri 5-10 ml/dk'ya düştüğünde ve hastaların diyaliz, böbrek nakli gibi böbrek replasman tedavilerine ihtiyaç duyulduğunda KBH olarak adlandırılır (Akpolat ve Utaş 2008).

Bu çalışmada, özellik seçim yöntemleri ile elde edilmiş niteliklere dayalı olarak sınıflandırma algoritmalarının performanslarının karşılaştırılması amaçlanmıştır. İlgili veri seti üzerinde özellik seçim yöntemleri uygulanarak en etkili yöntem(ler)in belirlenip literatürde popüler olarak kullanılan sınıflandırma algoritmaları çapraz doğrulama (cross validation) yardımı ile sınıflandırılmıştır. Sınıflama işlemi sürecinde doğru pozitif, yanlış pozitif, kappa istatistik, doğru sınıflandırma ve ROC (Receiver Operating Characteristics) eğrisi atında kalan AUC (Area Under the Curve) değerleri hesaplanmış olup ilgili değerler üzerinden karşılaştırılmıştır.

2. LİTERATÜR BİLGİLERİ

Çalışmanın bu bölümünde, Filtre (Korelasyon tabanlı özellik seçimi Ki-kare, F-skor, Relief-F, Bilgi Kazanımı, Simetrik Belirsizlik Kriteri, Kazanç Oranı, Tutarlılık Ölçütü), Sarmal (Ardışık İleri Yönde Kayan Seçim, Ardışık Geri Yönde Kayan Seçim, Bireysel En İyi Özellik Seçimi, l ekle – r çıkar seçimi, Ardışık İleri Yönde Kayan Seçim, Ardışık Geri Yönde Kayan Seçim, Genetik Seçim) özellik seçim yöntemleri ve sınıflandırma (Naive Bayes, k-En Yakın Komşu, Karar Ağacı, Rastgele Orman, Çok Katmanlı Algılayıcı, Radyal Temelli Fonksiyon Ağı, Destek Vektör Makineleri) algoritmalarına yer verilecektir.

2.1. Özellik Seçim Yöntemleri

ÖS, öznitelik seçimi olarak da bilinir, veri kümesinden en alakasız özellikleri çıkarmak ve ardından modelin daha iyi performansı için makine öğrenimi algoritmaları uygulama sürecidir. Çok sayıda alakasız özellik, eğitim süresini ve aşırı uyum riskini artırır (İnt. Kyn. 1).

(ÖS) yöntemleri, temel olarak mevcut veri seti ile ulaşılmak istenen en yakın tahmin değerine, hatta daha iyi tahmin değerlerine daha az özelliğe başvurarak ulaşmayı hedefleyen bir işlemler dizisidir. Daha az sayıda özelliği işleme dahil ederek daha iyi sonuç ile doğruluk, daha az maliyet ve işlem için sürenin kısalmasıyla daha hızlı sonuçlar elde edilir (Çifçi 2018).

Sınıflandırma problemlerinde ve birçok öğrenme algoritmalarında önemli bir yere sahip olan ÖS, konu ile ilgili olan, en faydalı ve önemli özelliklerin seçilerek veri kümesiyle ilgili özellik sayısının azaltılması için kullanılmaktadır. Bunun amacı, hesaplama yükünü azaltırken başarı oranını yükseltmektir. Özellik, nitelik veya değişken kavramları verinin görünümünden bahseder. Genellikle veriyi toplamadan önce, özellikler seçilir veya belirlenir (Koç 2016).

Molina vd. (2002)'ne göre ÖS; (i) mevcut özelliklerden başarı kriterini optimize eden bir alt kümenin seçilmesi, (ii) özelliklerin başarı kriterine göre en iyi belirli bir sayıda

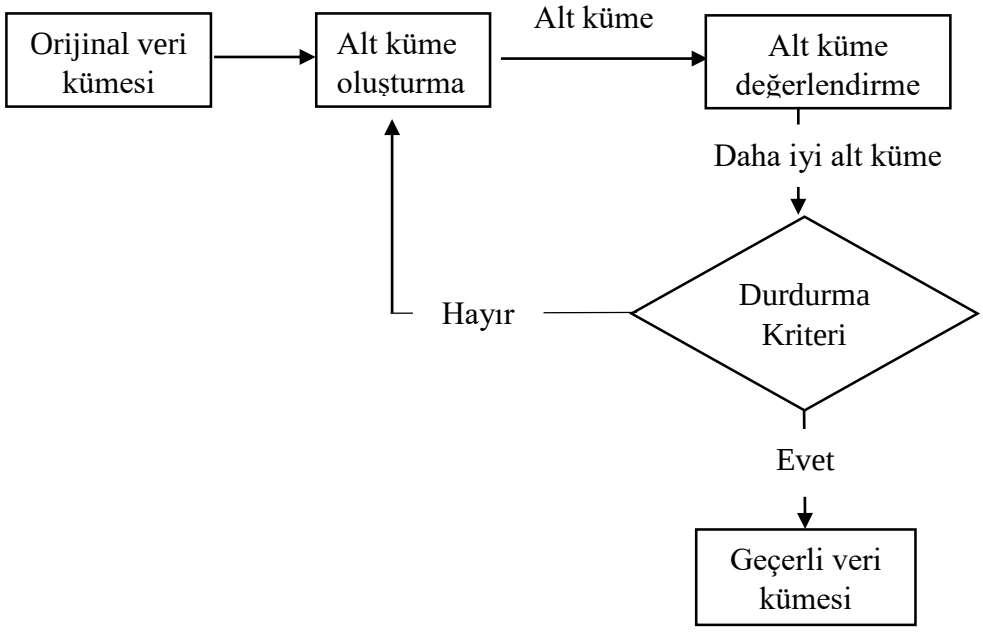
özelliğin seçilmesi ve (iii) seçilen özellik ile başarı kriteri arasında bir dengenin sağlanması olmak üzere üç şekilde tanımlanabilir (Beyazıt 2019).

ÖS'nin temel amacı, orijinal özelliklerin tamamını kullanmadan en yüksek oranda veri bütünlüğünü korumak ve aynı zamanda minimum özellik alt kümesini belirlemektir. Birçok gerçek dünya probleminde, ÖS gereksiz, yanıltıcı veya gürültülü verinin çokluğundan dolayı bir zorunluluk olarak düşünülmektedir. ÖS'den tamamıyla emin olabilmek için, tüm özellik alt kümelerinin (kombinasyonlarının) test edilmesi gerekmektedir (Koç 2016).

Dash ve Liu (1997)'ya göre tipik bir özellik seçim yönteminde dört temel adım vardır.

1) Bir sonraki aday alt kümesinin oluşturması için üretim süreci
2) İncelenen alt kümeyi değerlendirme süreci
3) Ne zaman duracağına karar vermek için bir durdurma kriteri
4) Alt kümenin geçerli olup olmadığını kontrol etmek için doğrulama süreci

Özellik Seçim sürecine ilişkin akış diyagramı şekil 2.1'de gösterilmiştir



Şekil 2.1 Özellik seçim süreci akış diyagramı (Dash ve Liu 1997).

Alt küme oluşturma süreci, hiçbir özellik olmadan yani boş kümeylede, tüm özelliklerle veya rastgele bir özellik alt kümesiyle başlayabilir. Değerlendirme işlevi, bazı üretim süreçleri tarafından üretilen bir alt kümenin iyiliğini ölçer ve bu değer, önceki en iyi alt küme ile karşılaştırılır. Uygun bir durdurma kriteri olmadan, özellik seçim süreci, alt küme uzayı boyunca kapsamlı veya sonsuza kadar çalışabilir. Üretim süreçleri ve değerlendirme fonksiyonları, bir üretim sürecine dayalı olarak bir durdurma kriteri seçimini etkileyebilir. Bunlar: önceden tanımlanmış sayıda özelliğin seçili olup olmadığı ve önceden tanımlanmış sayıda yinelemeye ulaşılıp ulaşılmadığıdır. Bir değerlendirme fonksiyonuna dayalı durdurma kriterleri; rastgele bir özelliğin eklenmesinin (ya da silinmesinin) daha iyi bir alt küme oluşturup oluşturmadığı ve bazı değerlendirme fonksiyonlarına göre en uygun bir alt kümenin elde edilip edilmediğidir. Döngü, bir durdurma kriteri sağlanana kadar devam eder (Dash ve Liu 1997).

Alt küme oluşturma, temelde değerlendirme için arama uzayındaki her bir duruma karşılık gelen bir üye alt kümenin belirlendiği sezgisel bir arama sürecidir. Bu süreç arama yönü üzerinde etkisi bulunan arama başlangıç noktasına veya noktalarına karar verir (Doak 1992; Uzer 2014).

2.1.1. Filtre Yöntemler

Makine öğrenimine ilişkin özellik seçiminde en eski yaklaşımlar filtre yöntemleridir. Tüm filtre yöntemleri, verilerin genel özelliklerine dayalı sezgisel yöntemler kullanır (Hall 1999). İlgilenilen veri seti için, algoritma önceden verilen boş, dolu veya rastgele seçilmiş bir alt kümeden aramaya başlar ve belirli bir arama tekniği ile özellik alanı içinde arama yapar. Üretilen her bir alt küme, bağımsız bir ölçüt ile değerlendirilip öncekiyle karşılaştırılır. Daha iyi olduğu tespit edilirse, mevcut en iyi alt küme olarak kabul edilir. Arama önceden tanımlanmış bir durdurma kriterine ulaşana kadar tekrar eder. Algoritma, o andaki son en iyi alt kümeyi çıktı olarak verir (Ezirmik 2020). Filtre yöntemi kullanılarak özellik seçimi bir kez yapılır ve daha sonra farklı sınıflandırıcılara girdi olarak sağlanabilir (Karegowda vd. 2010).

İsteğe bağlı bir filtreden geçirilen veriler üzerinde isteğe bağlı bir alt küme değerlendirici çalıştırma sınıfıdır (niteliklerin sırasını veya sayısını değiştiren filtrelere izin verilmez). Değerlendirici ve filtrenin yapısı yalnızca eğitim verilerine dayanmaktadır (İnt. Kyn. 9).

Arama stratejilerini veya değerlendirme ölçütlerini çeşitlendirerek farklı filtreleme algoritmaları tasarlanabilir. Filtre modelleriyle özellik seçimi yapmak için, bir özelliğin sınıflandırma süreciyle olan ilişkisini ölçmek amacıyla birkaç farklı ölçüt kullanılır. Tipik olarak, bu ölçütler özellik değerlerinin, özniteliğin farklı aralıkları üzerindeki dengesizliğini hesaplar (Ezirmik 2020). Bu model, sarmalayıcı yaklaşımından daha hızlıdır ve tümevarım algoritmasından bağımsız hareket ettiği için daha iyi bir genelleme sağlar (Marono vd. 2007).

2.1.1.1. Korelasyon Tabanlı Özellik Seçimi

Hall (1999)'a göre korelasyon tabanlı özellik seçimi, ilgilenen veri setine ait özellik alt kümelerinin bilgi değerlerini hesaplayan bir fonksiyon yardımıyla birlikte arama algoritmasınıda kullanmaktadır. Korelasyona tabanlı özellik seçimi, özellik alt kümelerini korelasyona dayalı sezgisel değerlendirme işlevine göre sıralayan basit bir filtre algoritmasıdır (Marono vd. 2007).

CFS'nin özellik altkümelere ilişkin değerleri hesaplarken kullandığı yaklaşım ilgili tüm özelliklerin sınıf etiketlerini tahmin etmekteki başarısı ile birlikte aralarındaki iç korelasyon değerlerini de baz almaktadır. Bu özellik seçim yöntemi, iyi özellik altkümeleri ilgilenen sınıf ile yüksek birbirleri ile düşük korelasyona sahip özelliklerden oluştuğu varsayımına dayanır. Sonuç olarak CFS yöntemi, diğer özelliklerle düşük korelasyonlu, sınıf değişkeni ile yüksek korelasyonlu olan özellikleri seçer. Yönteme ilişkin değer denklemi 2.1'deki gibi hesaplanır (Budak 2015).

$$M_s = \frac{k\bar{r}_{ci}}{\sqrt{k+k(k-1)\bar{r}_{ii}}} \tag{2.1}$$

Denklemdeki M_s, aşağıdakileri içeren bir özellik alt kümesi S'nin değeridir, k altkümeye ilişkin özellik sayısını, $\bar{r}_{ci}$ S ile ilgili özellik arasındaki ortalama korelasyonu, $\bar{r}_{ii}$

özelliklerin birbirleri arasındaki ortalama iç korelasyonunu göstermektedir (Budak 2015). Bu denklemin payı, sınıf kümesi özelliklerinin ne kadar öngörülebilir olduğunun bir göstergesi, paydasıda ne kadar fazlalığın özellikler arasında bulunduğunun bir göstergesi olarak düşünülebilir (Marono vd. 2007).

2.1.1.2. Ki-Kare Testi

Liu ve Setiono (1995)'a göre Ki-kare testi (X^2) iki değişken arasındaki ilişkinin bağımlı veya bağımsız olduğunu belirlemeye yarayan ayrık veriler için kullanılan bir hipotez test yöntemidir. Ki-kare istatistiğine dayalı özellik seçimi iki adım içermektedir. Yöntemin ilk kısmında özelliklerin sınıflara göre Ki-kare istatistikleri hesaplanır. İkinci kısımda serbestlik derecesi ve belirlenen önemlilik seviyesine göre Ki-kaynaşımı (Chi-merge) prensibi ile Ki-kare değerlerine bakılarak veri seti içerisindeki tutarsız özelliklerin bulunana kadar art arda özelliklerin ayrıştırılmasıdır (Şahin 2017). X^2 hesaplamasına ilişkin denklem 2.2'de verilmiştir.

$$X^2 = \sum_{i=1}^{2} \sum_{J=1}^{k} \frac{(A_{ij} - E_{ij})^2}{E_{ij}} \tag{2.2}$$

$$E_{ij} = \frac{R_i * C_i}{N} \tag{2.3}$$

Burada;

k = Sınıf sayısı

A_{ij}= Gözlenen değer

E_{ij}= Beklenen değer

R_i = i. satırdaki aralık

C_i= j. sütundaki sınıf

N = Anakütle Hacmi

X^2 istatistiğinin serbestlik derecesi, sınıf sayısının bir eksiğidir (Liu ve Setiono 1995).

2.1.1.3. F-skor Özellik Seçme Yöntemi

F-skor yöntemi, gerçek değerli iki sınıfın ayırt edilmesini ölçen basit bir tekniktir. F-skor yönteminde, veri kümesindeki her bir özelliğe göre f-skor değerleri hesaplanır ve hesaplanan f-skor değerlerinin ortalaması alınarak özellikleri seçmek için eşik değer seçilir. Eşik değerden büyük olan özellikler seçilir, diğerleri ise veri kümesinden uzaklaştırılır. $x_k = k = 1 \ldots m$ eğitme vektörleri verilsin, $n+$ ve $n-$ sırasıyla pozitif ve negatif örneklerin sayısı olsun. *i.* özelliğin f-skor değeri denklem 2.4'deki gibi hesaplanır (Polat 2008).

$$F(i) = \frac{(\bar{x}_i^{(+)} - \bar{x}_i)^2 + (\bar{x}_i^{(-)} - \bar{x}_i)^2}{\frac{1}{n_+ - 1}\sum_{k=1}^{n_+}(x_{k,i}^{(+)} - \bar{x}_i^{(+)})^2 + \frac{1}{n_- - 1}\sum_{k=1}^{n_-}(x_{k,i}^{(-)} - \bar{x}_i^{(-)})^2} \tag{2.4}$$

Burada, $\bar{x}_i, \bar{x}_i^{(+)}, \bar{x}_i^{(-)}$, sırasıyla tüm veri kümesi, pozitif ve negatif veri kümelerinin *i.* özelliğinin ortalamalarıdır. $x_{k,i}^{(+)}$, *k.* pozitif örneğinin *i.* özelliğidir ve $x_{k,i}^{(-)}$, *k.* negatif örneğinin *i.* özelliğidir. Pay, pozitif ve negatif kümeler arasındaki ayırımı gösterirken, payda ise pozitif ve negatif kümelerin varyanslarını gösterir. F-skor değeri ne kadar büyükse, o özelliğin ayırt edici özelliği de büyüktür. Fakat f-skor'nun bir dezavantajı, değişkenler arasındaki karşılıklı bilgiyi hesaba katmaz (Polat 2008).

2.1.1.4. Relief-F Algoritması

Orijinal Relief sözel ve sayısal özelliklerle ilgilenebilir. Ancak, eksik verilerle başa çıkamaz ve iki sınıflı problemlerle sınırlıdır. Bu ve diğer sorunları çözmesi, Relief-F algoritması olarak adlandırılır. Relief-F algoritması iki sınıf problemiyle sınırlı değildir, daha sağlamdır eksik ve gürültülü verilerle başa çıkabilir (Durgabai 2014).

Bu algoritma, denetimli öğrenme problemlerinde sonucu öngören özellikleri belirlemede üstündür ve özellikle standart özellik seçim algoritmaları tarafından normalde gözden kaçan özellik etkileşimlerini belirlemede iyidir. Relief-F algoritmalarının ana faydası, her bir ikili etkileşimi kapsamlı bir şekilde kontrol etmek zorunda kalmadan özellik etkileşimlerini tanımlamaları ve böylece kapsamlı ikili aramadan önemli ölçüde daha az

zaman almalarıdır (İnt. Kyn. 2). En yakın komşuların özelliklerinin değerleri sınıflandırılan örnekle karşılaştırılır ve her bir özellik için uygunluk puanlarını güncellemek için kullanılır. Bunun mantığı, faydalı bir özelliğin farklı sınıflardan örnekler arasında ayrım yapması ve aynı sınıftan örnekler için aynı değere sahip olmasıdır (Canedo vd. 2014).

2.1.1.5. Bilgi Kazanımı

Karar ağaçları, köklerden itibaren başlanarak yapraklara doğru yönlendirmeler ile oluşturulan ağaçlardır. Bilgi kazanımı (BK), karar ağaçlarının oluşturulmasında kullanılan önemli bir yöntemdir. Aslında kullanılan yöntemi entropi oluşturmaktadır. Entropi yardımıyla ulaşılan bu yönteme Bilgi Kazanımı denmektedir. Bilgi kazanımı, Veri seti içerisinde karara en çok etki eden özelliğin sayısal değerini bulmak için kullanılır. Bu yöntem entropi odaklı özellik seçimidir. $0 \leq H(Y) \leq 1$ aralığında değerler alan entropi 0'a yaklaştıkça belirsizlik azalır. Entropisi yüksek olan veri daha çok bilgi barındırır. Shannon'un Entropi yasası denklem 2.5'de verilmiştir (Akın ve Saraçlı 2014, Bulut 2017, Koşan vd. 2019).

$$H(Y) = -\sum_{y \in Y} p(y) \log_2 p(y) \tag{2.5}$$

Burada p(y), rastgele değişken Y için marjinal olasılık yoğunluk fonksiyonudur. S eğitim veri setinde gözlenen Y değerleri, ikinci bir X özelliğinin değerlerine göre bölündüğünde *X*'e uyan bölümlere göre Y'nin entropisi daha azdır, o halde Y ve X özellikleri arasında bir ilişki vardır. X'i gözlemledikten sonra Y'ye ilişkin entropi denklemi 2.6'de verilmiştir (Novakovic vd. 2011).

$$H(Y|X) = -\sum_{x \in X} p(x) \sum_{y \in Y} p(y \,|\, x) \log_2 (p(y|x)) \tag{2.6}$$

burada $p(y|x)$, x'i verilen y'nin koşullu olasılığıdır.

Bilgi kazanımı (BK) simetrik bir ölçüdür (Wyse vd. 1980; Novakovic vd. 2011). X'i gözlemledikten sonra Y hakkında edinilen bilgiler, Y'yi gözlemledikten sonra X hakkında

edinilen bilgilere eşittir. BK yöntemi, en ayırt edici özelliği seçme amacıyla kullanılır. BK'nın zayıf yönü, daha fazla bilgilendirici olmasalar bile daha fazla değere sahip özellikler lehine yanlı olmasıdır (Novakovic vd. 2011).
Y veri kümesine ilişkin, n tane alt bölüm *X* özelliğinden bölünecekse *X*'e ait bilgi kazancı hesaplanması denklem 2.7'da verilmiştir.

$$BK(Y,X) = H(Y) - H(Y|X) = H(X) - H(X|Y) \tag{2.7}$$

H(Y) veri kümesinin *X* üzerinden bölünmeden önceki Y'ye ilişkin, *H(X)* bölünmeden sonraki X'e ilişkin, $H(Y|X)$, X'i gözlemledikten sonra *Y*'ye ilişkin ve $H(X|Y)$ Y'i gözlemledikten sonra *X*'e ilişkin entropisini ifade eder (Novakovic vd. 2011).

Veri kümesinin bölünmeden önceki belirsizliğinin yüksek oluşu, veriye ilişkin bilgi verici özelliğin olduğunu göstermektedir. Bölünmeden sonraki belirsizliğinin düşük çıkmasıysa bu yöntemin veriyi dallara ayırma işlemini düzgün yaptığını göstermektedir (Yazıcı vd. 2015).

2.1.1.6. Simetrik Belirsizlik Kriteri

Kategori sayısı fazla olan değişkenlerde entropi değeri yüksek çıkma eğilimindedir. Simetrik Belirsizlik Kriterinde (SBK) özelliklerdeki kategori sayıları da dikkate alınarak bilgi kazancı ölçüsünde bir düzeltme işlemi uygular. SBK'ya ilişkin denklem 2.8'de verilmiştir (Press vd. 1992; Kocatürk vd. 2019).

$$SBK = 2\frac{BK}{H(X)+H(Y)} \tag{2.8}$$

Kazanım oranı ve simetrik belirsizlik ölçülerinin kullanılabilmesi için özelliklerin kategorik olması gerekir. Bu nedenle bu yöntemler uygulanmadan önce veri kesikli hale dönüştürülür (Kocatürk vd. 2019).

2.1.1.7. Kazanç Oranı

Kazanç Oranı (KO) bilgi kazancını aşağıdaki denklem 2.9'deki gibi normalleştirir (Quinlan 1993; Kuzey 2012).

$$Kazanç\ Oranı(Y,X) = \frac{Bilgi\ Kazancı(Y,X)}{Entropi(Y,X)} \tag{2.9}$$

Bilindiği üzere, bu oran payda sıfır olduğunda tanımsızdır. Ayrıca bu oran, payda çok küçük olduğu zaman özellikler lehine bir eğilim gösterir. Sonuç olarak, iki aşama önerilir. Öncelikle bilgi kazancı tüm özellikler için hesaplanır. Sadece, en az ortalama bilgi kazancı kadar performans gösteren özellikler baz alınır ve en iyi kazanç oranını elde eden özellik seçilir. Kazanç oranı hem doğruluk açısından hem de sınıflandırıcı karmaşıklığı açısından bilgi kazancı ölçütüne göre daha iyi performans göstermektedir (Quinlan 1988; Kuzey 2012).

2.1.1.8. Tutarlılık Ölçütü

Liu ve Setiono (1996)'ya göre Tutarlılık Ölçütü (TÖ), hedef kavramdaki tutarlılık düzeyine göre bir özelliği değerlendirir. Örnekler özellik alt kümesine yansıtıldığında, herhangi bir özellik alt kümesinin tutarlılığı, hiçbir zaman tam özellik kümesinden daha düşük olamaz (Zhao ve Liu 2007; Wang vd. 2013).

Bir özelliğin tutarlılık katkısı belirlenmiş bir değerden düşükse özellik kaldırılır aksi takdirde ilgili özellik seçilir. Bu yöntemin özellik etkileşimini yönetebileceği ve ilgili özellikleri verimli bir şekilde seçebileceği belirtilmiştir (Canedo vd. 2014). Eşit olan altkümedeki iki örnek olarak tanımlanır. Böylece amaç, sıfır tutarsızlığa yol açan minimum özellik alt kümesini bulmaktır (Molina vd. 2002).

Önerilen ölçü *U*, belirli bir özellik kümesi için veri kümesi üzerinde bir tutarsızlık oranıdır. Aşağıdaki açıklamadaki model, sınıf etiketi olmayan bir örneğin parçasıdır. Özellik alt kümesinin bir değer kümesidir. Sırasıyla $f_1, f_2, \dots, f_{|S|}$ özellikleri için

$n_{f1}, n_{f2}, \ldots, n_{f|S|}$ sayıda değer içeren bir özellik alt kümesi s için, en fazla $n_{f1} * n_{f2} * \ldots * n_{f|S|}$ model vardır. Tutarlılık ölçüsü, aşağıdaki gibi hesaplanan tutarsızlık oranı ile tanımlanır (Dash ve Liu 2003).

1) En az iki örnek varsa, model tutarsız olarak değerlendirilir. Sınıf etiketleri dışında hepsini eşleştirir. Örneğin, (0 1, 1) ve (0 1, 0) örneklerinin neden olduğu bir tutarsızlık burada iki özellik iki örnekte aynı değerleri alırken, örnekte son değer olan sınıf özelliği değişir.
2) Bir özellik alt kümesinin modeli için tutarsızlık sayısı, verilerde görünme sayısı ile farklı sınıf özellikleri arasındaki en büyük sayının çıkarılmasıyla elde edilen sayıdır. Örneğin, bir özellik alt kümesi *'s'* için, n_p örneklerinde bir *p* modelinin göründüğünü varsayalım. c_1,c_2,c_3 örneklerin sınıf etiketiyken $c_1+c_2+c_3 = n_p$ eşitliğini sağlar. c_3 üç sınıf arasında en büyüğüdür, tutarsızlık hesaplaması $(n-c_3)$. Burada dikkat edilmesi gereken husus, s özellik alt kümesinin verilerinde görünen farklı modeller p üzerindeki tüm n_ps'lerin toplamı veri kümesindeki toplam örnek sayısıdır (*P*), şöyleki $\sum_p n_p = P$
3) Bir özellik alt kümesinin $s(I_R(s))$ tutarsızlık oranı, tüm tutarsızlıkların toplamıyken, *P*'ye bölünen verilerde görünen özellik alt kümesinin tüm modellerini saymaktadır.

Tutarlılık durumu, $P \in P(\Omega, C)$ için, $X \subseteq \Omega$'un *P*'ye göre tutarlı olduğu kabul edilir. Bu durum denklem 2.10'da belirtilmiştir.

$$P(C = \xi \,|X = x) = 0{,}1 \tag{2.10}$$

x ile X arasındaki herhangi bir değer vektörü ve ξ sınıf niteliği için geçerlidir. Tutarlılık ölçüsü verilen olasılık dağılımı için negatif olmayan bir $\mu(P, X)$ değeri belirler $P \in \mathcal{P}(\Omega, C)$ ve bir özellik alt kümesi $X \in \mathfrak{P}_0(\Omega)$ sonucu olarak denklem 2.11'deki gibi ifade edilir.

$$\mu = \mathcal{P}(\Omega, C) * \mathfrak{P}_0(\Omega) \rightarrow [0, \infty) \tag{2.11}$$

Bu süreçte, mesafenin iki önemli özelliğine odaklanıyoruz. Birincisi, iki nokta arasındaki

mesafenin 0 olması, ancak ve ancak noktalar aynıysa bu durum söz konusu olur. Diğeri ise, iki nokta birbirine ne kadar yakınsa, mesafe değeri o kadar küçük olur. Burada, Ω sayılabilir anakütle özelliğini, $\mathfrak{P}_0(\Omega)$ Ω'nın tüm sonlu alt kümelerinin kümesini, $\mathrm{P}\ (\Omega, \mathrm{C})$ $\Omega \cup \{C\}$ üzerindeki olasılık dağılımları kümesini, X bir $\mathfrak{P}_0(\Omega)$ elemanını, C bir sınıf niteliğini temsil eden rastgele bir değişken olarak verilmiştir (Shin vd. 2011).
Bu ölçüm, tüm özelliklerde (sınıf özelliği dikkate alınmadan) aynı değerlere sahip tüm örnekleri (modelleri) bularak hesaplanan bir tutarsızlık oranı kullanır ve eşleşen tüm örneklerden her bir grup için aynı sınıfın en büyük örnek sayısını çıkartır. Seçilen tüm özellikler için aynı değerlere uyan örneklerin gruplandırılması, kendi gruplarının çoğunluk sınıfına ait olmayanlar için tutarlı örnekler aranması Liu'nun ölçüsüyle mümkündür, bu tutarsız örneklerin toplam örnek sayısı içindeki oranı olarak ifade edilebilir. Denklem 2.12'de verilmiştir (Azofra vd. 2008).

$$tutarsızlık = \frac{tutarsız\ örneklerin\ sayısı}{örnek\ sayısı} \tag{2.12}$$

Ölçüleri karşılaştırmak için tutarlılık ve tutarsızlık arasındaki ilişkiyi kurmak gerekir. Tutarlılık derecesini tutarsızlığın zıt değeri olarak tanımlanır. Tutarlılığa ilişkin denklem 2.13'de verilmiştir (Azofra vd. 2008).

$$tutarlılık = 1 - tutarsızlık \tag{2.13}$$

2.1.2. Sarmal Yöntemler

Bu metotlar bir öğrenme algoritması içerir. Öznitelik alt kümelerinin seçimi bir arama problemi olarak ele alınır ve seçimler diğer alt kümelerle karşılaştırılır. Seçilen özniteliklerin başarısı, bir öğrenme algoritmasında denenmektedir. Sarmal metotlar daha iyi doğruluk üretmelerine karşın, daha yüksek işlem maliyetine sahiptir (Çifçi 2018).

Literatürde Wrapper metot olarak adlandırılan bu yaklaşımda verinin karakteristiğine dair bilginin ve veri madenciliği algoritmasının bir araya gelmesiyle daha yüksek başarımlı modeller kurulduğu ve başarımı daha yüksek özellik seçimi yapıldığı söylenebilir. Ancak bu metotta her bir seçim hipotezi için model kurma işlemi gerçekleştirilir. Bu

modellerden başarımı en yüksek model seçildiğinde bu modele ilişkin özellik ağırlıklandırması ile özellik seçimi gerçekleştirilmiş olmaktadır. Başarım ölçütünün çok hassas olduğu iş modellerinde performans kaygısına rağmen bu yaklaşım tercih edilebilir (Beyazıt 2019).

2.1.2.1. Ardışık İleri Yönde Seçim

Ardışık ileri yönde seçim algoritması, Whitney (1971) tarafından önerilen basit ve etkin bir özellik seçim yöntemidir. Boş bir özellik kümesiyle işleme başlayarak, her bir adımda özellik altkümesinin ölçüt fonksiyonu değerini en iyi özellik altkümeye dahil edilir. Bu süreç, istenen özellik boyutuna ilişkin olası tüm sonuçlara ulaşıncaya kadar devam eder (Nasr vd. 2017; Bekiryazıcı 2020). Ardışık ileri yönde seçim algoritmasında, her etapta tek bir özellik altkümeye dahil edilir. Her etapta altkümeye *n* adet özelliğin eklendiği yöntem ise Genelleştirilmiş İleri Yönde Seçim'dir (Kittler, 1978). Her iki algoritmada içiçelik etkisine maruz kalmaktadır. Yani, seçilen özellikler bir kez kümeye dahil edildikten sonra bir daha kümeden çıkarılamaz. Bu sebepten dolayı iki algoritmada çoğunlukla alt en iyi sonuca ulaşabilmektedir. Ardışık ileri yönde seçim algoritması aşağıda verilmiştir (Günal 2008). Bu algoritma aşağıdaki gibi gösterilmektedir (Pratama vd. 2011; Budak 2015).

Boş özellik kümesi ile başla

$Y_0 = \{\emptyset\}$

1. Bir sonraki en iyi özelliği dahil et

   $x^+ = argmax_{x^+ \neq Y_k}[J(Y_k + x^+)]$

2. Eğer $J(Y_k + x^+) > J(Y_k)$ ise

2.1. $Y_{k+1} = Y_k + x^+$; k=k+1

3. Adım2'ye git
4. Dur

2.1.2.2. Ardışık Geri Yönde Seçim

Ardışık geri yönde seçim (AGYS) algoritması, ilk olarak Marill ve Green (1963) tarafından önerilmiştir. Bu algoritma, ardışık ileri yönde seçim algoritmasının tersi yönde çalışmaktadır. Başlangıçta özellik kümesinin tamamı göz önüne alınarak, her bir adımda o anki özellik altkümesinin ölçüt fonksiyonu değerini eniyileyecek şekilde bir öznitelik kümeden çıkarılır. Çıkarma işlemi, istenen öznitelik boyutuna ulaşıncaya kadar tekrarlanır. Her adımda bir yerine *n* adet özniteliğin elendiği yöntem ise Genelleştirilmiş Geri Yönde Seçim olarak isimlendirilir (Kittler 1978). İleri yönde seçim algoritmalarında olduğu gibi bu yöntemlerde de içiçelik etkisi söz konusudur. Seçim sürecinde, özellik(ler) kümeden bir kez çıkarıldıktan sonra bir daha giremez. Bu durum, yöntemlerin alt eniyi sonuç vermesine sebep olur (Günal 2008).

2.1.2.3. Bireysel En İyi Özellik Seçimi

Günal (2008)'a göre bu seçim yöntemi, tek değişkenli bir yaklaşımdır. Öznitelikler, belirlenen bir ölçüt fonksiyonuna göre bireysel olarak değerlendirilir ve sıralanır. İstenen sayıda özniteliğin seçiminde ise sıralı listedeki en önemli öznitelikten başlanıp sıradaki diğer özniteliklerle devam edilir. Bu yöntem oldukça hızlı olmasına rağmen öznitelikler arasındaki olası ilintileri değerlendirmediği için her zaman çok etkili olamayabilir. Öznitelik kümesindeki elemanların düşük ilintili ya da ilintisiz olması durumunda ise oldukça iyi sonuçlar alınabilmektedir (Uzer 2014).

2.1.2.4. l Ekle – r Çıkar Seçimi

l ekle - *r* çıkar yöntemi, ardışık ileri yönde seçim yönteminde kümeye seçilen bir özelliğin bir daha kümeden çıkartılamaması veya ardışık geri yönde seçim yönteminde kümeden çıkartılan bir özelliğin tekrar kümeye dahil edilememesi sorununun belli oranda giderilmesi amacıyla Stearns (1976) tarafından önerilmiştir. Yöntem temelde ardışık ileri yönde seçim ile ardışık geri yönde seçim yöntemlerinin birleşiminden oluşmaktadır. Algoritma, her adımda öncelikle ileri yönde seçim yöntemiyle *l* adet özelliği alt kümeye eklemekte ve daha sonra geri yönde seçim yöntemiyle *r* adet özelliği alt kümeden

çıkartmaktadır. İstenilen sayıda özelliği ulaşılıncaya kadar algoritma devam eder. *l* ekle - *r* çıkar özellik seçim yöntemine ilişkin algoritma aşağıda verilmiştir (Budak 2015).

1. Eğer $l > r$

1.1. ise boş özellik kümesi ile başla

$Y_0 = \{\emptyset\}$

1.2. değilse tam özellik kümesi ile başla

$Y_0 = X$

Adım 3'e git

2. *l* kez tekrarla (iyi özellik ekleme)

$x^+ = argmax_{x^+ \neq Y_k}[J(Y_k + x^+)]$

2.1. Eğer $J(Y_k + x^+) > J(Y_k)$ ise

$Y_{k+1} = Y_k + x^+$; k=k+1 olarak güncelle

3. r kez tekrarla (kötü özellik çıkarma)

$x^- = argmax_{x^- \neq Y_k}[J(Y_k + x^-)]$

3.1. Eğer $J(Y_k + x^-) > J(Y_k)$ ise

$Y_{k+1} = Y_k + x^+$; k=k+1

4. Adım2'ye git

5. Dur

2.1.2.5. Ardışık ileri yönde kayan seçim

Ardışık ileri yönde kayan seçim (AİYKS) yöntemi, *l* ekle - *r* çıkar yöntemine alternatif olarak Pudil vd. (1994) tarafından önerilmiştir. *l* ekle - *r* çıkar yönteminde yer alan *l* ve *r* değerleri belirlenirken herhangi bir teorik yapı kullanılmamaktadır. Bu nedenle, algoritmadan elde edilen sonuç belirlenen *l* ve *r* değerlerine bağlı olmaktadır. Bu sorunu giderebilmek adına ardışık ileri yönde kayan seçim algoritmasında *l* ve *r* değerlerini sabitlemek yerine kayan bir yapı kullanılmaktadır. Bu sayede, özellik seçiminin herhangi bir adımında mevcut sınıflama başarısı daha yüksek bir değere ulaşıncaya kadar aynı yönde hareket edilir. Bu yaklaşım, AİYKS algoritmasının birçok uygulamada AİYS algoritmasına göre daha başarılı sonuçlar vermesini sağlamakla beraber ciddi bir işlem

yükü de getirmektedir (Pudil vd. 1994, Budak 2015). Ardışık ileri yönde kayan seçim yöntemine ilişkin algoritma aşağıdaki gibidir (Pratama vd. 2011, Budak 2015).

1. Boş özellik kümesi ile başla

   $Y_0 = \{\emptyset\}$

2. Sıradaki en iyi özelliği seç

   $x^+ = argmax_{x^+ \neq Y_k}[J(Y_k + x^+)]$

3. Eğer $J(Y_k + x^+) > J(Y_k)$ ise

3.1. $Y_{k+1} = Y_k + x^+$; k=k+1 olarak güncelle

3.2. En kötü özelliği çıkar

   $x^- = argmax_{x^- \neq Y_k}[J(Y_k + x^-)]$

3.3. Eğer $J(Y_k + x^-) > J(Y_k)$ ise

3.3.1. $Y_{k+1} = Y_k + x^+$; k=k+1 olarak güncelle

3.3.2. Adım 3.2'ye git

3.4. Değilse

3.4.1. Adım2'ye git

4. Dur

2.1.2.6. Ardışık Geri Yönde Kayan Seçim

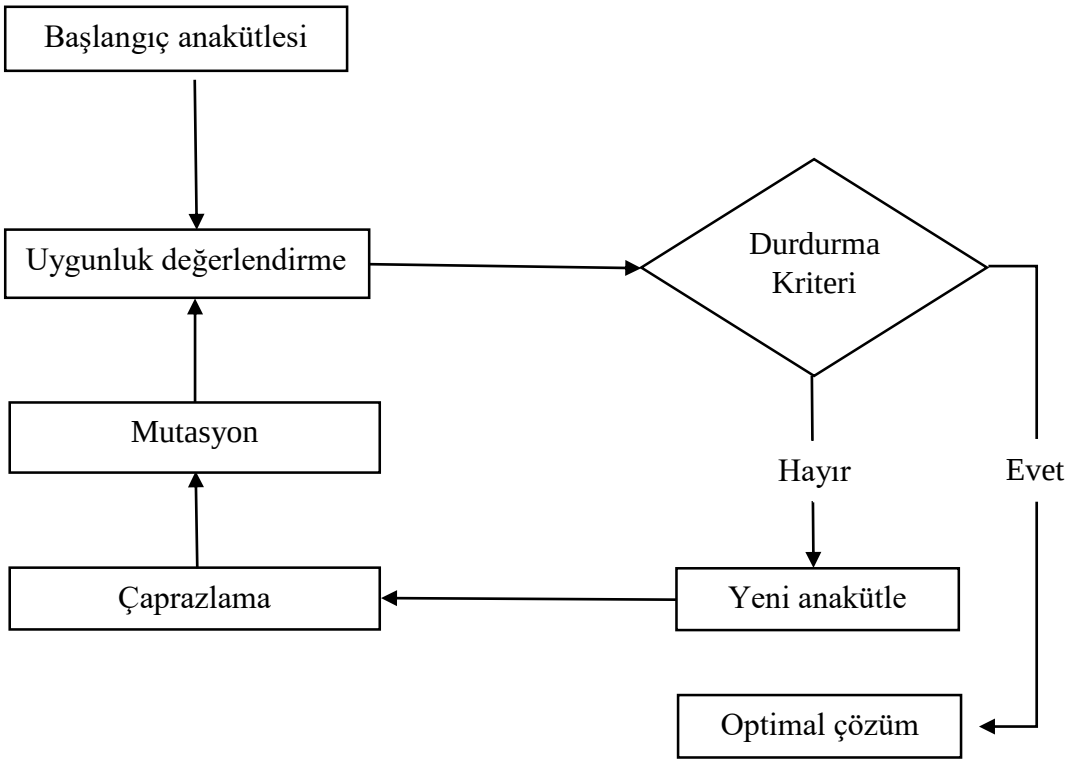
Ardışık geri yönde kayan seçim (AGYKS) yöntemi, Pudil vd. (1994) tarafından ardışık ileri yönde kayan yöntemi ile beraber önerilmiştir. Yöntem, açıklalan ardışık ileri yönde kayan yöntemi ile aynı prensiplere sahip olup, tersi yönde çalışmaktadır (Günal 2008).

2.1.2.7. Genetik seçim

Silahtaroğlu (2013)'na göre Genetik algoritma, doğadaki bilinen evrim yasalarından esinlenerek geliştirilmiş etkili bir algoritmadır. Veri madenciliğinde kümeleme, sınıflandırma, örüntü tanıma işlemlerinin yanı sıra özellik seçim işleminde de kullanılmaktadır. Genetik algoritmanın esinlendiği teoriye göre yaşayan canlılar hayatlarını devam ettirebilmeleri için diğer canlılar ile rekabet etmekte ve bu rekabet sonucunda başarılı olan genler bir sonraki kuşaklara aktarılmaktadır. Bu teori genetik algoritmanın çalışma prensibini oluşturmaktadır. Algoritma, nüfus olarak adlandırılan ve

kromozomlar tarafından temsil edilen bir dizi sonuçla işleme başlar. Eldeki bu sonuçlar (nüfus) kullanılarak yeni bir nüfus elde edilir. Teoriye göre elde edilen her yeni nüfusun bir öncekinden daha iyi olması beklenir. İstenilen durma kriterine ulaşılıncaya kadar benzer şekilde yeni nesiller üretilmeye devam edilmektedir (Budak 2015).

Evrimsel bir algoritma tasarlamanın ana arama bileşenleri şunlardır; genlerin sunumu, popülasyon başlangıcı, amaç fonksiyonu, seleksiyon, mutasyon ve çaprazlama ile yeniden üretim, nesillerin yer değiştirmesi ve durma kriteri (Ezirmik 2020).
Genetik algoritmaya ilşikin diyagram şekil 2.2' de verilmiştir.



Şekil 2.2 Genetik algoritma akış diyagramı (Ezirmik 2020).

2.2. Sınıflandırma Algoritmaları

Sınıflandırma algoritmaları, öğrenme algoritması temelli çalışır. Büyük verilerin içindeki gizli kalmış bir örüntüyü keşfetmek amacıyla uygulanır. Veri madenciliği kapsamında, örüntü, bir varlık için dijital ortamda kaydedilmiş; gözlemlenebilir, ölçülebilir ve tekrar

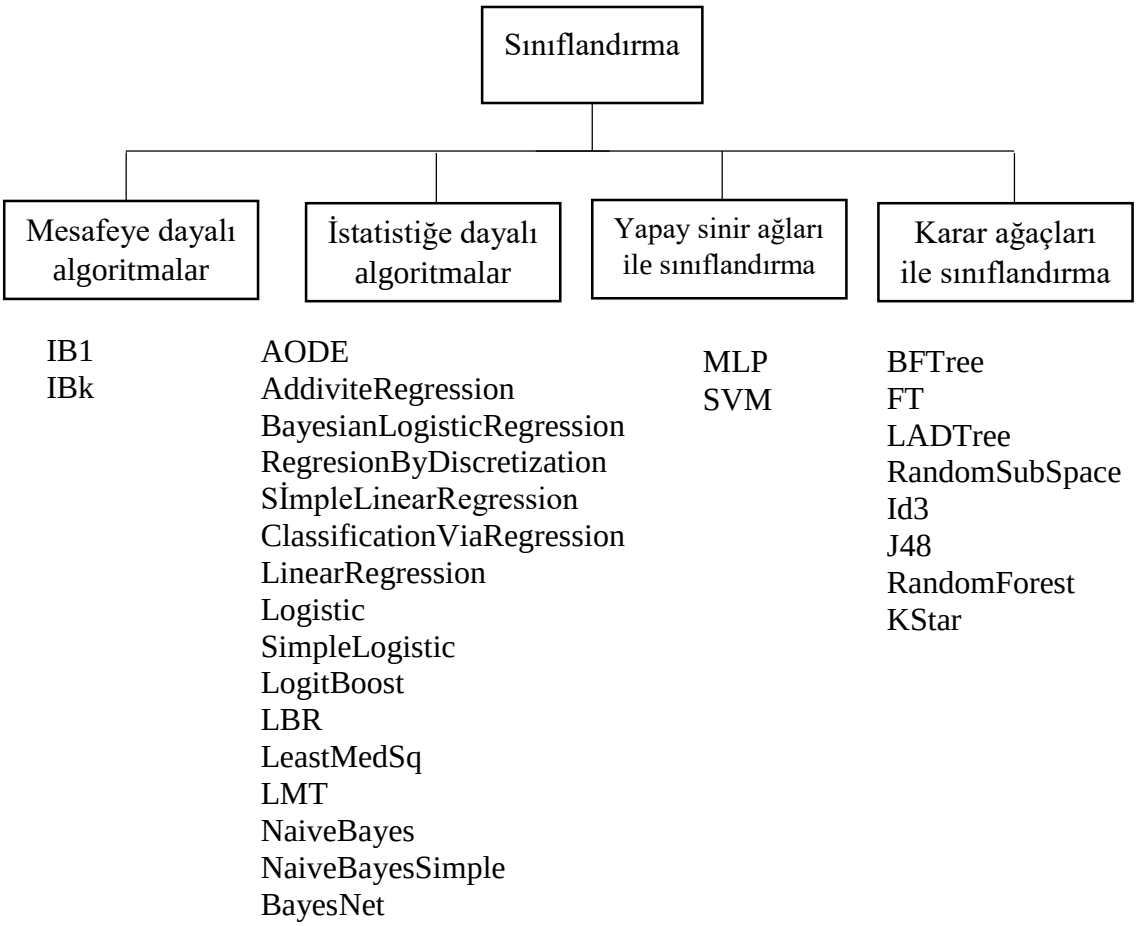
edilebilir bir bilgi olarak ifade edilmektedir. Ulaşılmak istenen bilginin elde edilmesi için uygulanan sınıflandırma algoritmaları, içerdiği verinin ortak özelliğine göre veri setinin belirli sınıflara ayrılmasını (ayrıklaştırılmasını) gerçekleştirir. Bu işlemin ardından bir sınıflandırma modeli elde edilir. Elde edilen sınıflandırma modeli yeni bir veri seti üzerinde uygulanarak, model ile belirlenmiş olan sınıfların veri seti içindeki benzerlerinin varlığı araştırılır (Çınar 2019).

Yapılan Literatür taramalarında en çok kullanılan sınıflandırma algoritmaları;

1. Lojistik regresyon
2. Linear diskriminant analizi
3. K-en yakın komsuluk tekniği
4. Sınıflama ve regresyon ağaçları
5. Bagging
6. Boosting
7. Random Forest
8. Destek vektör sistemleri
9. Yapay sinir ağları
10. Nearest shrunken centroids
11. Naive Bayes

Sınıflandırma kavramı temel olarak bazı belirli kurallara göre bir veri kümesinde tanımlanan sınıflar arasında veri dağıtma olarak tanımlanabilir. Literatürde birçok sınıflandırma yöntemi bulunmaktadır. Burada önemli olan veri kümesine göre doğru sınıflandırma algoritmasını belirlemek ve kullanılan algoritmanın başarı oranının yüksek olması gerekmektedir.

Bu çalışmada mesafeye (k- En yakın komuşuluk), İstatistiğe dayalı (Naive Bayes), Yapay sinir ağlarına (Çok Katmanlı Algılayıcı, Destek Vektör Makinaları, Radyal Temelli Fonksiyon Ağı) ve rastgele orman ile karar ağaçlarına dayalı sınıflandırma algoritmalarına yer verilmiştir. Sınıflandırma algoritmalarına ilişkin Weka'nın hiyerarşik yapısı şekil 2.3'de gösterilmiştir.



Şekil 2.3 Weka'nın hiyerarşik yapısı (Tapkan vd. 2011)

2.2.1. Naive Bayes

Naive Bayes sınıflandırıcısı, tahminler arasında bağımsızlık varsayımıyla İngiliz matematikçi Thomas Bayes'in adını verdiği Bayes Teoremine dayanan bir sınıflandırma tekniğidir. Basit bir ifadeyle, bir sınıftaki belirli bir özelliğin varlığının diğer herhangi bir özelliğin varlığı ile ilgisiz olduğunu varsayar. Bu özellikler birbirine veya diğer özelliklerin varlığına bağlı olsa bile, bu özelliklerin tümü bağımsız olarak olasılıklara katkıda bulunur. Naive Bayes modelinin oluşturulması kolaydır ve özellikle çok büyük veri kümeleri için kullanışlıdır. Sadeliği ile birlikte, Naive Bayes'in oldukça karmaşık sınıflandırma yöntemlerinden bile daha iyi performans gösterdiği bilinmektedir (Ezirmik 2020).

Bayes sınıflandırıcısı, güçlü bağımsızlık varsayımları ile Bayes teoreminin uygulanmasına dayanan basit bir olasılıksal sınıflandırıcıdır. Altta yatan olasılık modeli için daha açıklayıcı bir terim, bağımsız özellik modeli olacaktır. Basit terimler, bir Naive Bayes sınıflandırıcısı, bir sınıfın belirli bir özelliğinin varlığının veya yokluğunun, başka herhangi bir özelliğin varlığı veya yokluğu ile ilgisi olmadığını varsayar (Vafeiadis vd. 2015). Basit sınıflandırma algoritmaları kategorisinde yer almakta ve dengesiz sınıflı verilerde de çalışmaktadır. Algoritmanın çalışma prensibi, bir eleman için her durumun olasılığını hesaplamakta ve olasılık değeri en yüksek olana göre sınıflandırılmaktadır (Choubey vd 2017; Demiraslan ve Suner 2021).

Naive Bayes, Bayes teoreminden faydalanılarak oluşturulmuş sınırlandırma için kullanılan anlaşılabilir ve kolaylıkla uygulanabilir en basit makine öğrenme algoritmalarından biridir. Bu yöntemle bir örneğin hedef niteliğinin sınıf değerine ait olma olasılığı bulunabilmektedir. Bayes teoremi denklem 2.14'de verilmiştir.

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)} \tag{2.14}$$

Eşitlikte *X*; özellik vektörü, *H* ise bir özellik vektörünün *C* gibi bir sınıfa ait olma olasılığını ifade eden hipotezdir. *P(H|X)* ise koşullu olasılığı temsil eder. Bayes teoremi göz önüne alındığında Naive Bayes sınıflandırıcısının algoritması ise şu şekildedir;

Naive Bayes; Bayes ağı yapısal bir model ve bir dizi koşullu olasılıklardan oluşur (Jiang vd. 2009). Genellikle sınıflandırma algoritmalarında kullanılır.

Bayes sınıflandırma algoritması;

Veri kümesini al: D

Sınıf Alın: C1, C2, . . , n

R : sınıflandırılacak kayıt

Her C için yap

Denklem 2.15'ü kullanarak hesapla

Döngü

En yüksek değere sahip C'ye R atama

Durdur

D'nin veri setini temsil ettiği ve *D*'deki her *X*'in sınıf etiketinin belirli olduğu varsayılsın. *X*, *n* tane öznitelikten oluşan bir vektördür ve $X=(x_1, x_2, \ldots x_n)$ olarak temsil edilmektedir. $C_1, C_2, \ldots Cm$ ile temsil edilen *m* tane sınıf olduğu varsayılsın. Naive Bayes sınıflandırıcısı bir *X* vektörünün *Ci* sınıfına ait olup olmadığını bulmak için, bütün sınıflar içinde en yüksek *P(Ci |X)* koşullu olasılığa sahip değeri bulmaya çalışır. Bu durum Bayes teoremi ile denklem 2.15'de ifade edilmiştir (Kaynar vd. 2017).

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)} \tag{2.15}$$

P(X) değeri tüm sınıflar için aynı olduğundan, yalnızca *P(X|Ci)P(Ci)* ifadesi maksimum yapılmalıdır. *P(Ci)* ifadesi, *Ci* sınıfındaki eleman sayısının, tüm eleman sayısına oranıdır. *P(X|Ci)* ifadesi ise, *X*'in n tane değer içeren bir öznitelik vektörü olduğu varsayıldığında aşağıdaki denklem 2.16 ile hesaplanır.

$$P\left(X\middle|C_{i_i}\right) = \prod_{k=1}^{n} P(X_k|C_i) \tag{2.16}$$

Sonuçta, sınıflandırıcı en büyük *P(X|Ci)P(Ci)* ifadesine sahip olan *Ci* örnek uzaydaki sınıf değerini, *X* vektörünün örnek uzayı olarak seçer (Kaynar vd. 2017).

Naive Bayes algoritmasında, olasılık değerleri temel olarak karar noktasında etkilidir. Doğal olarak, diğer algoritmalarda da bir olasılık vardır. Bununla birlikte, buradaki olasılık değerlerinin koşullu olasılıklar olduğunu bilmekte fayda var. Bu yönüyle Naive Bayes algoritmasının diğer algoritmalardan farklı olduğu yer tam olarak burasıdır.

2.2.2. k En Yakın Komşu Algoritması

k-En Yakın Komşu (k-NN) algoritması, ilk olarak 1950'lerin başında ortaya atılmıştır (Han vd. 2011; Dilki ve Başar 2020). Mitchell (1997)'e göre k-NN, en temel örnek tabanlı öğrenme algoritmalarından biridir. Örnek tabanlı öğrenme algoritmalarında, öğrenme işlemi eğitim veri setinde tutulan verilere dayalı olarak gerçekleştirilmektedir. Yeni karşılaşılan bir örnek, eğitim veri setinde yer alan örnekler ile arasındaki benzerliğe göre sınıflandırılmaktadır (Taşçı ve Onan 2016).

k-NN, en yakın verileri desen alanında arayarak bilinmeyen örnekleri sınıflandırmak için sınıflandırma yöntemidir (Galit vd. 2010). k-NN, aşağıdaki gibi tanımlanan denklem 2.17'de ifade edilen öklid mesafesini kullanarak sınıfı tahmin eder:

$$d(x,y) = \sqrt{\sum_{i=1}^{k}(x_i - y_i)^2} \tag{2.17}$$

Öklid uzaklığı d(x,y), desen uzayındaki en yakın örnekleri bulma mesafesini ölçmek için kullanılır. Bilinmeyen örneğin sınıfı, komşularından gelen çoğunluk oyu ile tanımlanır.

k-NN yaygın olarak kullanılan bir makine öğrenme algoritmasıdır. k-NN, iyi performansı yanı sıra basitliği sebebiyle yaygın olarak kullanılmaktadır (Yıldız 2019). Seçilen bir özelliğin kendine en yakın olan özellikle arasındaki yakınlığı kullanarak sınıflandırma yapılır (Kılınç vd. 2016). Başarılı bir sınıflandırma için k-NN üç önemli faktöre bağlıdır; Bunlar k değeri, komşuları belirlemek için kullanılan mesafe metriği ve örnek boyutudur (Yıldız 2019). Büyük k değerleri seçildiğinde aşırı uyuma neden olacağından dolayı dikkat edilmesi gerekmektedir (Aydın 2018). k-NN sınıflandırma algoritması, eğitim veri setindeki nokta ve noktalar arasındaki mesafeleri hesaplar. Literatürde k-NN algoritmasının k değeri için farklı formüller yer almaktadır. Genel denklem 2.18'daki gibidir (Filiz vd. 2017).

$$f(x) = argmax_{c \in C} = \sum_{i=1}^{k} w_i \delta(c, f(x_i)) \tag{2.18}$$

kıyaslar.nBu süreçte tahminler, komşu örneklerine ilişkin oy çoğunluğuyla belirlenmektedir. Ölçüm için öklid uzaklığı kullanılmaktadır.

k-NN algoritmasında, daha önce elde edilmiş ve sınıfı önceden belirlenmiş başka bir veri tarafından sınıflandırılacak verilerin yakınlığı incelenir ve ardından bir sınıflandırma işlemi uygulanır. Algoritmanın temel mantığı, hangi komşuya yakın olursa olsun, o gruba benzediği varsayımını tutmasıdır. Burada önemli olan kaç komşunun incelenmesi gerektiğini bilmektir. Bu, algoritmanın en önemli değişkenlerinden biri olan komşu sayısını verir.

2.2.3. Karar Ağaçları Algoritması

En önemli sınıflama algoritmalarından biri olan karar ağaçlarında, öğrenme algoritması basittir. Açığa çıkarılan bilginin gösterimi kolaylıkla anlaşılabilir. Karar ağaçları yalnızca kararları göstermezler, aynı zamanda kararların açıklamasını da içerirler (Emel ve Taşkın 2005). Quinlan (1993)'a göre karar ağacı, en yüksek bilgi kazancına sahip özellik aracılığıyla verilerin bölünmesiyle oluşturulan düğümlerden bir karar sonucuna ulaşılır (Akçetin ve Çelik 2014).

Karar ağaçları, basit yorumlanmasının yanı sıra veri tabanına kolayca dahil edilmeleri, yüksek güvenirliğe sahip olmasından ötürü sınıflamandırma algoritmaları içerisinde en çok başvurulan algoritmadır. Bu algoritma tahmin edici ve tanımlayıcı özelliklere sahiptir (Köktürk 2012).

Karar ağacı, sınıflandırmayı anlamak ve yorumlamak için kolay bir yöntemdir (Xing vd. 2007). Karar ağacı yöntemi sınıflandırma algoritmalarında en popüler algoritmalardan biridir. Bu yöntem entropiye dayanır. Karar ağaçlarının inşasında en önemli nokta, hangi değişkenin ilk döngü, yani kök döngü olduğunu belirlemektir (Atılgan 2011).

Bir karar ağacı oluştururken izlenecek adımlar;

• Veri kümesini al: D

• Sınıf al: C

• Sayım numarası alanlar: f

• Başla: C sınıfı için genel durum entropiyi denklem 2.19 yardımıyla hesaplayın

$$H(D) = -\sum(p_y \log_2 p(y) \tag{2.19}$$

Her f için yap

Ayırma bilgilerini denklem 2.20 ile hesapla

$$H(\frac{|D_1|}{|D|}, \dots, \frac{|D_s|}{|D|}) \tag{2.20}$$

Kazanç Oranını Hesapla

Kazanç oranı (D; S) = Kazanç(D=S) Ayırma Bilgileri (D; S)

• Döngü

• En Küçük Kazanç Oranına Sahip Değişken Düğümler Atayın.

• Durdurma Kriteri = DOĞRU yaprak ise DUR

• Değilse

Başlaya git.

Örnek karar ağacı gösterimi şekil 2.5'de verilmiştir.

Burada *BD* karar düğümünden çıkan tüm karar dalları arasındaki en yüksek beklenen değerdir. Buna göre beklenen değer hesaplamaları genellikle aşağıda sıralanan kurallar uyarınca gerçekleştirilir (Gordon ve Pressman 1983; Lezki 2014).

1) Bir karar düğümüne bağlanan bitiş dalı için BD, sonuca eşittir.
2) Bir şans düğümüne bağlanan bitiş dalı için BD, bu dalın sonucu ile olasılığının çarpımıdır.
3) Bir şans düğümü için BD, her bir şans dalının sonucu ile bunlara karşılık gelen olasılıklarının çarpımlarının toplamıdır.
4) Bir karar düğümü için BD, karar düğümünde çıkan tüm karar dallarının beklenen değerleri içinde en büyük kazanç değerine sahip olanıdır.
5) Herhangi bir düğümün BD, başlangıç düğümü yönünde bağlantılı olduğu bir önceki düğümün sonuç değeridir.

Standart karar ağaçları, örnek alanının ayrık bölgelere bölünmesi için çalışır. Karar ağaçlarını öğrenmek için algoritmaların çoğu, parçalardan birini yinelemeli olarak ikiye bölerek çalışır. Her parça en fazla bir kez bölünebilir. Başka bir deyişle, yalnızca yaprak düğümleri bölünebilir. Genel olarak değişen karar ağaçlarında her parçanın birden çok kez bölünmesine izin verir (Yoav ve Yahni 1999).

Eğitim verilerine ait öznitelik bilgilerinden yararlanılarak bir karar ağacı yapısı oluşturulmasında temel prensip verilere ilişkin bir dizi sorular sorulması ve elde edilen cevaplar doğrultusunda hareket edilerek en kısa sürede sonuca gidilmesi olarak ifade edilebilir. Bu şekilde karar ağacı sorulara aldığı cevapları toplayarak karar kuralları oluşturur. Ağacın ilk düğümü olan kök düğümünde verilerin sınıflandırılması ve ağaç yapısının oluşturulması için sorular sorulmaya başlanır ve dalları olmayan düğümler ya da yapraklar bulunana kadar bu işlem devam eder (Kavzaoğlu ve Çölkesen 2010).

Karar ağacı tekniğini kullanarak verinin sınıflanması, öğrenme ve sınıflama olmak üzere iki basamaklı bir işlemdir. Öğrenme basamağında önceden bilinen bir eğitim verisi, model oluşturmak amacıyla sınıflama algoritması tarafından analiz edilir. Öğrenilen model, sınıflama kuralları veya karar ağacı olarak gösterilir. Sınıflama basamağında ise test verisi, sınıflama kurallarının veya karar ağacının doğruluğunu belirlemek amacıyla

kullanılır. Eğer doğruluk kabul edilebilir oranda ise kurallar, yeni verilerin sınıflanması amacıyla kullanılır. Eğitim verisindeki hangi alanların hangi sırada kullanılarak ağacın oluşturulacağı belirlenmelidir (Çalış vd. 2014).

2.2.4. Rastgele Orman Karar Ağacı

Rasgele orman veya rasgele karar ağaçları topluluğu (ensemble learning), öğrenme yöntemleri sınıflandırma ve diğer görevler için kullanılır. Rasgele karar ağaçları, karar ağaçlarının aşırı uyum davranışını çalışma zamanında (training time) düzeltmektedir. Algoritma İlk olarak Ho (1998) tarafından, rasgele altuzay örnekleme, Eugene Kleinberg tarafından öne sürülen "stokastik ayrımcılık" yaklaşımıyla ele alınan sınıflandırmayı çözmeyi amaçlar (Ho 1998; Kesenek 2019).

Breiman (2001)'a göre Rastgele Orman karar ağacı, veri setinde en iyi niteliklerden seçilen düğümleri dallara ayırmak yerine, her bir düğümde rastgele alınan niteliklerin en iyisini seçerek tüm düğümleri dallara ayırır. Her veri kümesi asıl veri setinden yer değiştirmeli olarak üretilir. Rastgele özellik seçimi kullanılarak ağaçlar geliştirilir ve budama işlemi yoktur. Rastgele orman algoritmasının diğer algoritmalara göre daha hızlı ve doğru olmasının sebebi bu yöntemdir (Akçetin ve Çelik 2014). Rastgele orman algoritmasında sonradan dahil edilen veriye ilişkin tahmin yapılmasının yanı sıra, değişkenlere ait önem derecesi de hesaplanmaktadır. Veri setin bünyesinde çok sayıda değişken barındırıyorsa, değişkenlerin önem derecelerinin hesaplanmasıyla değişken sayısını azaltma yönünden etkilidir (Akman 2010). Bu süreçte birden fazla karar ağacı oluşturarak doğru sınıflandırma oranını yükseltmesinde etkili bir algoritmadır. Rastgele olarak seçilen karar ağaçlarının birleşmesiyle karar ormanı oluşturulur. Veri setlerinde bulunan fazla sayıdaki değişken ve sınıf etiketine sahip kategorik değişken bulunduran, veri setlerinde iyi sonuçlar verir (Aydın 2018).

Algoritmada ağaç yapısının oluşturulması için her bir düğümde kullanılacak örneklerin sayısı ve oluşturulacak ağaç sayısının belirlenmesi gerekir. Sınıflandırma sırasında karar ormanı, kullanıcı tarafından belirlenen K adet ağaçtan oluşturulur. Yeni bir nesne sınıflandırılacağı zaman bu K adet karar ağacı tarafından işleme tabi tutulur ve her

ağaçtan elde edilen oranlar içerisinde en yüksek olanı seçilerek sınıf belirlenmesi yapılır (Pal 2005, Çölkesen 2009, Karakoyun ve Hacıbeyoğlu 2014).

Rastgele orman modelinin kurulmasında aşağıdaki aşamalar izlenir (Akman 2010).

• Orijinal veri setinden, öğrenme veri seti ayrıldıktan sonra, öğrenme veri setinden sınıf dağılımına uygun şekilde rastgele örneklemler seçilir.

• Oluşturulan örnek veri setinden kullanıcının belirlediği sayı kadar değişken ağaç yapısında kullanılmak üzere rastgele seçilir. Eğer orijinal veri setinde, sınıf değişkeni hariç, toplam değişken sayısı M ise, ağaç yapısında analistin belirlediği R tane değişken kullanılacaktır. Ağacın çok fazla büyümemesi ve aşırı öğrenme sorunu oluşmaması için ağaç yapısında bütün değişkenler kullanılmamaktadır. Burada $R<M$ olmak zorundadır.

• Karar ağacı oluşturulurken her düğümde, belirlenen R tane değişkenden dallara ayrılmaya bilgi kazancı en yüksek değişkenden başlanılır. Dallara ayrılacak değişken belirlendikten sonra her düğümden aşağıya doğru iki dal oluşturulur. Dalların hangi değere göre ayrılacağına gini indeksi kullanılarak karar verilir. Bu işlem her düğüm için yeni oluşturulacak dal kalmayıncaya kadar tekrar edilir.

• En düşük hata oranına sahip ağaca en yüksek, en yüksek hata oranına sahip olan ağaca en düşük ağırlık verilir. Bu şekilde kurulan tüm ağaçlara, ağacın hata oranının değerine göre göreceli olarak bir ağırlık verilir.

• Ormanın sınıflaması yapılırken, her ağaç oluşturduğu sınıflardan birine ağırlıklı olarak oy verir. Orman, veri setindeki her deneğe ait sınıf tahminini, tüm ağaçların yaptığı tahminlerin bir araya getirilmesi neticesinde, ağırlıklı olarak en çok oyu almış sınıfı seçerek yapar.

• Rassal Orman yönteminde bireysel olarak oluşturulan ağaçlar üzerinde budama işlemi yapılmaz. Her ağaçta kullanılan veri ve değişkenler farklı olduğundan, Rassal Orman yöntemi aşırı öğrenmeye ve gürültülü veriye karşı güçlüdür.

Mather (2005)'e göre Rastgele orman algoritmasında, diğer karar ağaçlarına benzer şekilde, dallanma kriterlerinin belirlenmesi ve uygun bir budama yönteminin seçilmesi önemlidir. Dallanma kriterlerinin belirlenmesinde Gini Katsayısı yöntemi kullanılmaktadır (Karakoyun ve Hacıbeyoğlu 2014).

Shang vd. (2007)'ne göre Gini indeks özellik seçimi çalışmalarında kirlilik/safsızlık (nonpurity) ölçülmesi esasına dayanır. Veri kümesi içerisindeki özellik kirlilik ölçüsünün az olması uygun değer özelliği, tam tersi durumda safsızlık ölçüsü yüksek olan özellik ise etkisiz özelliği temsil eder. Ancak Gini indeks teorisi üzerine yapılan çalışmalarda saflık ölçüsüne bakılır, yani saflık değeri yüksek olan özellik en iyi özellik olarak kabul edilir. Gini indeksi algoritmasının ana fikri aşağıdaki gibidir (Şahin 2017). S terimi s örneklerinden oluşan küme varsayılır. Bu örneklemler m kadar farklı sınıfa sahiptir. Sınıfların farklılıklarına göre, S kümesi m kadar alt sınıfa bölünebilir. Si örneği Ci'ye ait örnek seti olduğu varsayılırsa, si ise Si'e ait örnek sayısıdır (Şahin 2017).

Bir özelliğin Gini değeri hesaplanmadan önce özelliğin lk sol ve sağ değerleri hesaplanmaktadır. Bu hesaplamalar Denklem 2.21 ve 2.22'deki gibi hesaplanır.

$$Gini_{sol} = 1 - \sum_{i=1}^{k} \left[\frac{L_i}{|T_{sol}|} \right]^2 \tag{2.21}$$

$$Gini_{sağ} = 1 - \sum_{i=1}^{k} \left[\frac{R_i}{|T_{sağ}|} \right]^2 \tag{2.22}$$

Burada k sınıfların sayısını, T bir düğümdeki örnek sayısını, T_{sol} ve $T_{sağ}$ kollardaki örneklerin sayısını, L_i ve $R_i$ise kollardaki i. kategorisindeki örneklerin sayısıdır (Adak ve Yurtay 2013).

Hesaplanan bu değerler Gini değerinin hesaplanmasında kullanılır. Denklem 2.23' de verilmiştir.

$$Gini_j = \frac{1}{n}(\left| T_{sol} \right| Gini_{sol} + \left| T_{sağ} \right| Gini_{sağ}) \tag{2.23}$$

İlgilenen her bir özellik için hesaplanan en küçük Gini değeri seçilir (Adak ve Yurtay 2013).

ÇKA'nın temel yapısı en az 3 katmandan oluşur. İlk katmana giriş katmanı, ikinci gizli katman ve son katmana çıktı katmanı denir. Ayrıca ÇKA veri kümesinin eğitiminde geri yayılma kontrollü bir öğrenme tekniği kullanır. Öğrenme prosedürü, girdiler ve istenen çıktılar hakkındaki bilgisine dayanarak gizli birimlerin iç parametrelerini belirlemek zorundadır. Bu nedenle öğrenme, çok büyük bir parametre alanını aramaktan oluşur ve bu nedenle genellikle oldukça yavaştır.

Çok katmanlı algılayıcı ağ modeline ilişkin denklem 2.24'de yer alan eşitlikle ifade edilmektedir.

$$Y_n = f_0\{b_0 + \sum_{k=1}^{h}[w_k * f_n(b_{hk} + \sum_{i=1}^{m} w_{ik}x_{ni})]\} \quad (2.24)$$

Denklem 2.24'da yer alan değişkenlere ilişkin açıklamalar aşağıdaki gibidir (İşeri ve Arıman 2019).

Y_n**:** Normalleştirilmiş çıktılar
f_0: Çıktı katmanı transfer fonksiyonu (bu çalışmada sigmoid kullanılmıştır)
b_0: Eşik Değer, bias
w_k: k. gizli katman ile çıkış katmanı arasındaki bağlantı ağırlığı
f_n: Gizli katmana ilişkin transfer fonksiyonu
b_{hk}: k. gizli katmanın bias terimleri
w_{ik}: Girdi katmanında bulunan i. nörondan gizli katmanda bulunan k. nöron arasında bağlantı ağırlığı
x_{ni}: Normalleştirilmiş girdi vektörü

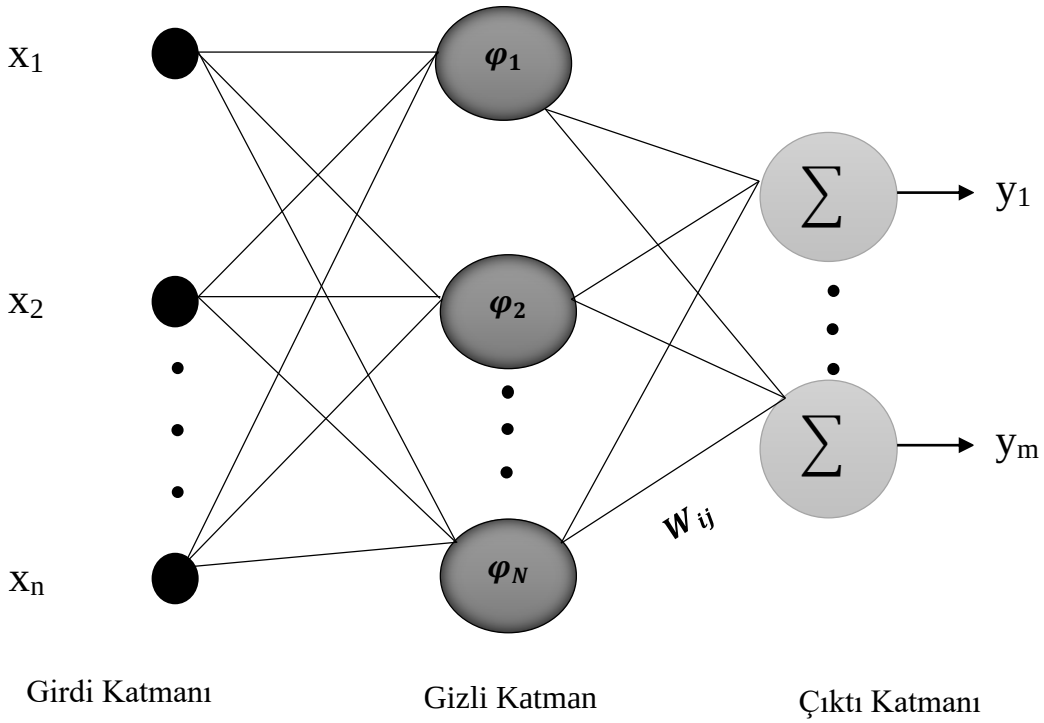
ÇKA aslında yapay bir sinir ağıdır. Yapay sinir ağlarından farkı, daha fazla katmana sahip olmasından kaynaklanmaktadır. Bu fazla katmanlar sayesinde nöronlar arasındaki sinyaller herhangi bir kayıp olmadan daha doğru ve sorunsuz bir şekilde iletilebilir. Algoritmanın önemli değişkenleri arasında katman sayıları, öğrenme hızı ve aktivasyon fonksiyonu olduğu söylenebilir. Aktivasyon fonksiyonundaki değişiklikler yöntemin başka bir sürümü olarak görünür. Bu fonksiyon radyal tabanlı bir fonksiyon olarak alındığında, karşımıza RTF Ağı çıkmaktadır.

2.2.6. Radyal Temelli Fonksiyon Ağı

Radyal temelli fonksiyon (RTF) yaklaşımı, köklerini Powell (1987)'ın çalışmasından alır. Yapay Sinir Ağlarında (YSA) öğrenmeye alternatif bir araç olarak kullanımı, düzensiz konumlandırılmış veri noktaları göz önüne alındığında çok değişkenli enterpolasyon için özellikle uygundur. YSA kullanmaları geleneksel çok katmanlı algılayıcılar üzerinde kendi avantajları nedeniyle çeşitli mühendislik alanlarında sınıflandırma sorunları, fonksiyon yaklaşımı, gürültülü aradeğerleme ve çözüm uygulamalarına düzenleme buldu yani daha hızlı yakınsama, küçük tahmin hataları ve daha yüksek güvenilirlik (Ke'gl vd. 2000, Kagoda vd. 2010, Ababaei 2012).

RTF ağı tasarımı ise çok boyutlu uzayda eğri uydurma yaklaşımıdır ve bu nedenle RTF'nin eğitimi, çok boyutlu uzayda eğitim verilerine en uygun bir yüzeyi bulma problemine dönüşür. RTF'nin genellemesi ise eğitim sırasında bulunan çok boyutlu yüzeyin kullanılmasına eşdeğerdir. RTF, sayısal analizde çok değişkenli problemlerin çözümünde kullanılmış ve YSA'nın gelişmesi ile birlikte bu fonksiyonlardan YSA tasarımında yararlanılmıştır. RTF, ileri beslemeli YSA yapılarına benzer şekilde giriş, saklı ve çıkış katmanından oluşur ancak, giriş katmanından gizli katmana dönüşüm, radyal tabanlı aktivasyon fonksiyonları ile doğrusal olmayan sabit bir dönüşümdür. Gizli katmandan çıkış katmanına ise doğrusal bir dönüşüm gerçekleştirilir. RTF'de uyarlanabilecek serbest parametreler; merkez vektörleri, radyal fonksiyonların genişliği ve çıkış katman ağırlıklarıdır (Şenol 2010).

RTF ağı yöntemi genellikle modelleme alanında zaman serileri, sınıflandırma sorunları, sistem kontrolleri tahminlerinde kullanılır. YSA ile hemen hemen aynı işlev verir. YSA'dan tek farkı, radyal temel işlevini bir aktivasyon fonksiyonu olarak kullanmasıdır.



Şekil 2.7 Radyal temelli ağ yapısı (Okkan ve Dalkılıç 2012).

RTF'in matematiksel ifadesi denklem 2.25'de ifade edilmektedir.

$$\boldsymbol{Y_i(x) = \sum_{i=1}^{N} W_{ij} \varphi_i(x)} \tag{2.25}$$

Burada $Y_i(x)$ *i.* radyal temelli fonksiyon, W_{ij} gizli katmandaki *i.* nörondan çıkış katmanındaki *j.* nöron arasındaki ağırlığı, *N* ise gizli katmanda bulunan hücre sayısını, $\varphi_i(x)$ ise aktivasyon fonksiyonunu gösterir. RTF'de gizli katman aktivasyon fonksiyonu olarak genellikle Gauss Fonksiyonu kullanılır. Gauss Fonksiyonu *x* giriş vektörünü c_i merkezi $\|x - c_i\|$ uzaklığını, σ_i de standart sapma değerini simgelemekte olup denklem 2.26'de ifade edilir (Şenol 2010).

$$\varphi_{(i)}(x) = e^{\frac{\|x-c_i\|}{2\sigma_i^2}} \tag{2.26}$$

Moradkhani vd. (2004)'ne göre RTF ağda gizli katman düğüm sayısı en çok giriş veri seti adedi kadar olabilir. Gizli katmandaki farklı düğüm sayısı ile yapılan hesaplama ayrı bir çözümdür. Görüldüğü gibi RTF ağında parametre ve hesaplama işlemi az olduğundan,

hesaplama süresi de kısa olur. Dolayısıyla fazla kaynak kullanmadan, çok hızlı bir şekilde eğitilebilir. RTF tekniği iyi bir genelleme olanağı, daha az düğüm kullanma ve kısa hesaplama olanağı sunar (İlkuçar 2015).

2.2.7. Destek Vektör Makineleri

Kapasite kontrol prensibi (KKP) 1970'lerin ortalarında keşfedilmiş olsa da, bu ilke yeni algoritmaların geliştirilmesine yol açmıştır. 1990'larda destek vektör makineleri (DVM) olarak adlandırılmaya başlandı. Son birkaç yılda makine öğrenimi teorisindeki en etkili gelişmelerden biri Vapnik'in DVM üzerindeki çalışmalarıdır (Vapnik 1982).

DVM, sınıflandırma ve regresyon analizinde, verileri analizinde ve örneklerden öğrenen denetimli bir öğrenme yöntemi olarak kullanılmaktadır (Bilişik 2011). DVM'nin temelleri Vapnik (1995) tarafından atılmıştır. Vapnik (1995), farklı sınıfların nesnelerine ait kümeler arasında, öznitelik uzayında en iyi bir hiper düzlem oluşturan bir sınıflandırıcı geliştirmiştir (Kaya 2014). DVM, veriyi birbirinden ayıran en uygun doğrusal ya da doğrusal olmayan bir fonksiyon yardımıyla sınıflandırma yapmaktadır (Yıldız vd. 2012). Guyon vd (2002)'ne göre Destek vektör makineleri ve özyinelemeli nitelik eleme yapılarının birlikte kullanıldığı bir algoritmadır (Kaya 2014).

Buradaki temel amaç onları destek vektörleri olarak gruplamaktır. Bu gruplandırmalar doğrusal veri kümeleri için daha uygundur. Ancak, veri kümeleri doğrusal olmayan veri kümelerindeki çekirdek işlevlerinin yardımıyla doğrusallaştırılabilir ve uygulanabilir. Bu çalışmada Poly Kernel, Normalleştirilmiş Poly Kernel, Puk ve Radyal Temelli Fonksiyon (RTF) Çekirdek fonksiyonları kullanılmıştır. DVM, en iyileştirme sorununu çözerek hiper düzlem bulur. Bu durum denklem 2.27 yardımıyla gerçekleştirilir.

$$maxQ(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{j=1}^{n} \alpha_i \alpha_j d_i d_j x_i^T x_j \qquad (2.27)$$

Burada $0 \leq \alpha_i \leq c \;\; için \quad i = 1,2, \dots, n.$

DVM, çıktıyı hesaplamak için çekirdek işlevi biçiminde tanımlanan *f(x)* karar işlevine

2.8'de iki boyutlu düzlemde birbirinden doğrusal ayrılabilen iki sınıfa ait verilerin dağılımı ve bu verilerin birbirinden çok sayıda doğru ile ayrılabildiği görülmektedir. Çok boyutlu uzayda veriler hiper düzlemler ile ayrılabilir (Özkan 2008; Yıldız vd. 2012).

Verileri birbirinden ayıran en uygun hiper düzlem, birbirine en uzak iki hiper düzlem bulunarak elde edilir. Şekil 2.8'de birbirine en uzak H_1 ve H_2 düzlemleri arasından geçen H_0 hiper düzlemi, iki sınıfı birbirinden ayıran en uygun hiper düzlem olarak seçilmektedir. H_0 düzlemine en uygun ayırma hiper düzlemi adı verilir. H_1 ve H_2 düzlemleri üzerindeki her bir veri destek vektörü olarak adlandırılır (Yıldız vd. 2012). Destek vektör algoritması en büyük sınır genişliğine sahip ayırıcı hiperdüzlem ile sınıflandırma yaparak eğitim hatasını minimize etmeye çalışır. Sınıfları birbirinden ayıran marjini en büyük, doğrusal bir ayırt edici fonksiyon bulunmasını amaçlar (Küçük vd. 2013).

Veri setlerini en iyi ayırma işlemini matematiksel olarak gösterecek olursak, bu verileri ayıran karar sınırı denklemi $w^Tx+b=0$ şeklinde ifade edilir. Bu karar çizgisine paralel olan sınır çizgilerinin negatif hiper düzlem ve pozitif hiper düzlem denklemleri sırasıyla $w^Tx+b=-1$ ve $w^Tx+b=1$ olur. Buradan iki sınır çizgisi arasındaki geometrik uzaklık hesaplanacak olursa $m = \frac{2}{\|w\|}$ denklemi yardımıyla bulunur. Burada amaç bu geometrik aralığı en büyük yapmaktır (Demirçalı 2015).

Doğrusal olmayan DVM, ilgili veri setinin doğrusal bir fonksiyon ile tam ya da belirli bir hata ile ayrılamaması söz konusu olduğunda başvurulan bir algoritmalardır. Gercek yaşam problemlerinde bir veri setinin hiper düzlemle doğrusal olarak ayrılması çoğunlukla mumkun değildir. Bu bağlamda sınıfları ayırma işlemi, ayırma eğrisinin tahmin edilmesiyle mumkun olmaktadır. Ancak uygulamada eğrinin tahmin edilmesi oldukça zordur (Ayhan ve Erdoğmuş 2014). Sınıflandırmak istenilen veri seti şekil 2.9'daki gibi doğrusal bir çizgi ile ayrılamayacak durumda olabilir (Karakoyun ve Hacıbeyoğlu 2014).

$$w^T\phi(t) = \langle w, \phi(t) \rangle = 0 \tag{2.29}$$

Burada $w^T = [w_1 \dots w_m]$ vektöründeki w_i'ler karar eğrisini belirleyen katsayılardır, $\phi(t)$ vektörü de özellik uzayında $\phi(t) = [\phi_1 \phi_2 \dots \phi_m]^T$ şeklinde m boyutlu bir vektördür (Topaloğlu 2014).

DVM'lerinde Polinomiyal çekirdeği, tüm eğitim verilerinin normalleştirildiği problemler için çok uygundur. Polinomiyal çekirdek fonksiyonu denklem 2.30'te ki gibi ifade edilir.

$$K(x, y) = (\alpha x^T y + c)^d \tag{2.30}$$

Burada α eğimi, c sabit terimi ve d polinom derecesini ifade etmektedir (Güldoğan 2017).

DVM algoritmasında kısacası, sınıflandırmalar yapılırken limitler yardımıyla bir karar alınır. Bu algoritmadaki temel mantık, verileri bir doğru boyunca ayırmaktır. Veriler bir doğru boyunca ayrılamazsa, eğriler yardımıyla ayrılır. Çekirdek işlevleri bu eğrilerin hesaplanmasında kullanılır. Bunun bu çalışmada kullanılan en zor algoritmalardan biri olduğu yorumlanabilir.

Pearson VII işlevi, Karl Pearson tarafından 1895'te geliştirildi ve genellikle X-ışını kırılma taramalarının ve Kızılötesi spektrumlardaki tek bantların eğri uydurması için kullanılır (Pearson 1895; Üstün vd. 2005). Eğri uydurma için Pearson VII fonksiyonunun(PUK) genel formu denklem 2.31'de verilmektedir (Zhang ve Ge 2013).

$$f(x) = \frac{H}{\left[1+\left(\frac{2(x-x_0)\sqrt{2^{\frac{1}{\omega}}-1}}{\delta}\right)^2\right]^{\omega}} \tag{2.31}$$

burada H, tepe noktasının x_0 merkezindeki tepe yüksekliğidir ve x, bağımsız değişkeni temsil eder. Parametreler ve ω, tepe noktasının yarı genişliği ve kuyruk faktörüdür. Bu bağlamda, bir işlev ancak ve ancak şu durumlarda geçerli çekirdek işlevleri sınıfına aittir:

karşılık gelen çekirdek matrisi simetrik ve pozitif yarı tanımlıdır. PUK'nin gerçekten bu koşulları sağladığını göstermek için, Uestuen iki vektörün bir fonksiyonunu denklem 2.32' deki gibi ifade etmiştir (Uestuen vd. 2006; Zhang ve Ge 2013).

$$K(x_i, x_j) = \frac{1}{\left[1+\left(\frac{\sqrt{\|x_i - x_j\|^2 \sqrt{\frac{1}{2\omega}-1}}}{\sigma}\right)^2\right]^{\omega}} \tag{2.32}$$

Burada x_i ve x_j iki vektör değişkenidir (Zhang ve Ge 2013).

Genel olarak polinom çekirdeği denklem 2.33'deki gibi tanımlanır;

$$K(x_i, x_j) = (a + x_i^T x_2)^b \tag{2.33}$$

Burada b çekirdeğin derecesini ve a ise sabit terimi ifade etmektedir (İnt. Kyn. 7).

RTF Kernel DVM sınıflandırma algoritmasında kullanılan varsayılan çekirdektir ve denklem 2.34'deki gibi ifade edilir

$$K(x_i, x_j) = e^{-\varphi \|x_i - x_j'\|^2} \tag{2.34}$$

Burada $\|x_i - x_j'\|^2$ öklid uzaklığını, φ tek bir eğitim örneğinin etkisinin ölçüsünü belirtir (İnt. Kyn. 8).

Her ne kadar veriye bağımlı kerneller güçlü bir alternatif olarak son zamanlarda popüler olsalar da, RTF kerneller çok daha popülerdir. RTF kerneller için yakınsama polinominal kernellerden daha yavaş olmasına rağmen RTF kerneller çoğu zaman daha iyi performans gösterirler (Tolun 2008).

3. MATERYAL ve METOT

Bu bölümde, çalışmada kullanılan veri setinin açıklanması, uygulanan özellik seçim yöntemleri ve sınıflandırma algoritmaları detaylandırılmıştır**.**

Veriler 2015-07-03'te Hindistan'da bulunan Apollo Hastanelerinden toplanmıştır. Toplam 400 kişiye ilişkin bilgiler içermektedir. Bu 400 kişinin 250'sinin Kronik Böbrek Hastalığı varken, geri kalan 150 kişinin Kronik Böbrek Hastalığı bulunmamaktadır. Bu veriler 24 değişken değer ile birlikte kan ve idrar analizi sonucu elde edilmiştir. Bir kişinin Kronik Böbrek Hastalığından muzdarip olup olmadığıyla ilgili bir sınıflandırma değişkeni mevcuttur. Yani toplamda 25 değişken olup bu değişkenlerin 11'i sayısal, 14'ü nominaldir. Verilere (İnt. Kyn. 4)'dan ulaşılmıştır. Ayrıca, ilgili bağlantıdan indirilen verilerde bazı eksik değerler de vardır. Veri kümesinin ayrıntıları çizelge 3.1'de gösterilmiştir. Ayrıca, tüm sınıflandırma süresince 10 kat çapraz doğrulama kullanılmıştır. Kullanılacak programa göre excel'deki mevcut verilere yalnızca dosya uzantıları ayarlanmıştır. Bu süreçte verilerin yapısına zarar verecek bir değişim yapılamamış olup ayrıca, ilgili veri kümesinden hiçbir değişken çıkarılmamıştır.

Çizelge 3.1 Kronik böbrek hastalığı veri setinin açıklaması

Ozellik1	Yaş	Ozellik14	Potasyum
Ozellik2	Kan basıncı	Ozellik15	Hemoglobin
Ozellik3	Ozgül ağırlık	Ozellik16	Paketlenmiş hücre hacmi
Ozellik4	Albumin	Ozellik17	Beyaz kan hücresi sayımı
Ozellik5	Şeker	Ozellik18	Kırmızı kan hücresi sayımı
Ozellik6	Kırmızı kan hücreleri	Ozellik19	Hipertansiyon
Ozellik7	Iris hücresi	Ozellik20	Şeker Hastalığı
Ozellik8	Iris hücre kümeleri	Ozellik21	Koroner arter hastalığı
Ozellik9	Bakteriler	Ozellik22	Iştah
Ozellik10	Kan şekeri	Ozellik23	Ayak ödemi
Ozellik11	Kandaki üre miktari	Ozellik24	Kansızlık
Özellik12	Serum kreatinin	Özellik25	Sınıf
Ozellik13	Sodyum		

Çalışmanın uygulama kısmında weka paket programında bulunan özellik seçim yöntemlerinin tamamı ilgili veri setine uygulanmış sonuç olarak korelasyon, filtre ve tutarlılık yöntemleri daha az sayıda özellik(değişken) ile ilgili veri setini açıklayabilmiştir. Çalışmada kullanılan toplam 25 adet değişken olup korelasyon tabanlı özellik seçim yöntemi uygulandığında 16 değişken ile, filtre özellik seçim yöntemi uygulandığında 11 değişken ile, tutarlılık özellik seçim yöntemi uygulandığında ise 4

değişken ile açıklanmıştır. Burdan hareketle çalışmada kullanılan sınıflandırma yöntemlerinin performansları bir özellik seçiminin olmadığı (değişkenlerin tamamının kullanıldığı) durum için ve bahsi geçen bu üç özellik seçim yöntemi için hesaplanmıştır.

Son olarak, tüm sınıflandırma algoritmaları 16, 11, 4 ve 25 değişkenin tümü değerlendirmeye alınmıştır**.** Doğru sınıflandırma oranları bu değişken sayılarınca yapılmıştır.

Bu çalışmada kullanılan makine öğrenimi algoritmaları genel anlamda karşılaştırıldığında, Bu algoritmalardaki temel mantık, sınıf doğrultusunda maksimum sayı ile tahmin yapmaktır. Bu kadar basit algoritmaların çalışmaya dahil olmasının temel nedeni, en basit algoritmanın sonucunun ne olduğunu bulmak ve diğer algoritmalarda başarı elde edilip edilmediğini belirlemek için tasarlanmış olmasıdır. Aslında, diğer algoritmalar biraz daha kapsamlıdır ve birkaç işlemden geçtikten sonra işlev görür.

Bu deneysel çalışmada KBH veri setine ilişkin, Karar ağacı, k-NN (k=2), ÇKA, Naive Bayes, RTF ağı ve DVM (Poly Kernel, NormalizePoly Kernel, Puk ve RTF Kernel sınıflandırma algoritmaları kullanılmıştır.

ROC analizi, testin gücünü ayırt etme yeteneğinin belirlenmesinde, çeşitli test tekniklerinin karşılaştırılmasında ve uygun pozitif eşiğin belirlenmesinde kullanılır. ROC analizi sınıflandırma algoritmalarının sonuçlarını değerlendirmek için kullanılan bir yöntemdir (Takıcı 2018). Eğri Altındaki Alan (AUC), ROC eğrisinin altındaki alanı ifade eder. Bu alan 1'e ne kadar yakınsa, tanı oranının o kadar yüksek olduğu anlamına gelir.

Bir sınıflandırma probleminin doğruluğu, en yüksek evrensel değerlendirme ölçülerinden biridir ve denklem 3.1 'de verlmiştir (DP - Doğru Pozitif, DN - Doğru Negatf, YP -Yanlış Pozitif, YN – Yanlış Negatif). Bu önlemin faydası, test senaryolarının mutlak sayısından uygun şekilde sınıflandırılmış test senaryolarının sayısını bulabilmesidir (Rahman vd. 2020).

$$Doğruluk\ (\%) = \frac{|DP+DN|}{|DP+DN+YP+YN|} \tag{3.1}$$

Kappa testi, iki ve daha fazla gözlemciye ait uyumlarına ilişkin güvenilirliğini ölçen testdir. (Congalton ve Green 1998; Aydın 2018). Eğer gözlenen değerler uyum şansa bağlı olarak uyumdan büyük ya da eşit ise, $\kappa \geq 0$; gözlenen uyum şansa bağlı olarak uyumdan daha küçük ise, $\kappa < 0$ olmaktadır. $\kappa = 1$ olması durumunda tam uyum gerçekleşir. Kappa katsayısının yorumlanabilir aralığı $0 \leq \kappa \leq 1$ arasındadır ve negatif ($\kappa < 0$) durumu söz konusu olduğunda güvenilirlik açısından anlamlı değildir. Kappa değeri 0,4'ün üzerinde olması istenen durumdur. Kappa değeri denklem 3.2'deki gibi hesaplanır (Aydın 2018):

$$\kappa = \frac{(P_0 - P_c)}{(1 - P_c)} \tag{3.2}$$

Burada; P_0 kabul edilen oran, P_c beklenen oranı göstermektedir.

Landis ve Koch (1977)'a göre, elde edilen κ değerlerini yorumlaması çizelge 3.2'de sunulmuştur (İnt. Kyn. 3).

Çizelge 3.2 Kappa değerlerinin yorumlanması

κ	Yorum
≤ 0	Uyum yok (Şansa bağlı olabilecek uyumdan daha kötü uyum olması)
0.1-0.20	Önemsiz uyum
0.21-0.40	Düşük derecede uyum
0.41-0.60	Orta derecede uyum
0.61-0.80	İyi derecede uyum
0.81-1.00	Mükemmel uyum

Öte yandan Kappa değerine ilişkin sonuçların kategorik sayısından da etkilendiği unutulmamalıdır. Kategori sayısı ne kadar küçük olursa hesaplanan kappa değeri de o kadar büyük olmaktadır. Bir başka dikkat edilecek husus ise incelenecek durum çok nadir karşılaşılan bir durum ise uyumun göstergesine ilişkin bulunan Kappa değeri de küçük olmaktadır (Viera ve Garrett 2005; Kılıç 2015).

Çapraz doğrulama bir tahmin modelinin bağımsız bir veri setinde ne kadar iyi performans gösterebileceğini değerlendirmek için güçlü bir yöntemdir. Çapraz doğrulama, temel eğitim verilerinin tahmine dayalı potansiyelini, tahmini sapmadan dahili olarak test edilmesine olanak tanır. Temel süreç basittir: verileri rastgele birkaç eşit alt kümeye bölüür, ardından yinelemeli olarak tahmine dayalı modeller oluşturulur ve test edilir, öyle ki alt kümelerin her biri bir kez alıkonulur ve bir kez model testi için kullanılırken kalan alt kümeler modeli eğitmek için kullanılır (İnt. Kyn.5).

K-katlı çapraz doğrulamada, orijinal örnek rastgele k eşit boyutlu alt örneğe bölünür. k alt örnekten, modeli test etmek için doğrulama verisi olarak tek bir alt örnek tutulur ve kalan k-1 alt örnekleri eğitim verisi olarak kullanılır. Çapraz doğrulama işlemi daha sonra k kez tekrarlanır (katlar), her k alt örneğin doğrulama verisi olarak tam olarak bir kez kullanılır. Kıvrımlardan elde edilen k sonuçları daha sonra tek bir tahmin üretmek için ortalaması alınabilir (veya başka bir şekilde birleştirilebilir). Bu yöntemin avantajı, tüm gözlemlerin hem eğitim hem de test için kullanılması ve her gözlemin doğrulama için tam olarak bir kez kullanılmasıdır. Sınıflandırma problemleri için, tipik olarak, her katın kabaca aynı oranlarda sınıf etiketi içereceği şekilde katların seçildiği tabakalı k-kat çapraz doğrulama kullanılır (İnt. Kyn. 6).

4. BULGULAR

Birçok çalışmada KBH'nın çeşitli algoritmalar aracılığıyla en yüksek doğruluk oranıyla tahmin etmesi amaçlanmıştı. Tüm bu çalışmalar, birçok algoritma ve veri kümesi kullanarak hangi algoritmanın en doğru sonucu verdiğini belirlemek için çaba sarf edildi. Bu çalışmada kullanılan özellik seçim yöntemleri neticesinde sınıflandırma algoritma sonuçlarımız ile benzer çalışmalarda kullanılan algoritmalar ve sonuçları çizelge 3.5'te karşılaştırılmıştır. Ayrıca, bu çalışmaların karşılaştırmalarını yapabilmek için elbette aynı koşullarda analiz etmek gerekir. Bu 5 çalışmada kullanılan veri kümeleri ve değişkenler aynıdır. Ancak, diğer yazarlar kullandıkları algoritmaların değişkenleri hakkında ayrıntılı bilgi vermedikleri için, sınıflandırma algoritmalarının bazıları aynı sonuçları verirken, bazıları farklı sonuçlar sunmuştur. Bu çalışmayı diğer 4 çalışmadan üstün kılan durumlar, ilgili veri seti üzerinde özellik seçim yöntemleriyle indirgenen veri üzerinden çapraz geçerlilik yardımıyla sınıflama yapılmıştır. Ayrıca, bu çalışma diğer yazarların kullanmadığı bazı sınıflama algoritmalarını da içermektedir. Bu sayede daha fazla algoritma kullanılmış ve en yüksek sınıflandırma oranına sahip algoritmayı belirleme şansı elde edilmiştir.

Çizelge 3.3 Farklı Özellik seçimleri ve sınıflandırma algoritmaları için DP, YP, ROC ve Kappa istatistikleri

	Özellik Seçim Yok				**Filtre**				**Korelasyon**				**Tutarlılık**			
	Doğru pozitif	Yanlış pozitif	ROC	KAPPA	Doğru pozitif	Yanlış pozitif	ROC	KAPPA	Doğru pozitif	Yanlış pozitif	ROC	KAPPA	Doğru pozitif	Yanlış pozitif	ROC	KAPPA
NaiveBayes	0.92	0	1	0.89	0.96	0	1	0.94	0.95	0	1	0.93	0.93	0	0.99	0.91
k-NN(k=2)	0.94	0	0.97	0.92	0.99	0	0.99	0.99	0.97	0	0.98	0.96	0.96	0.04	0.97	0.96
Karar Ağacı	0.99	0.02	0.99	0.97	0.99	0.03	0.99	0.96	0.99	0.02	0.99	0.97	0.98	0.04	0.99	0.95
Rassal Orman	1	1	1	1	0.99	0	1	0.98	0.99	0	1	0.99	0.98	0	0.99	0.97
RTF Ağı	0.97	0	0.99	0.96	0.99	0.07	0.99	0.98	0.97	0	0.98	0.96	0.98	0.01	0.99	0.96
ÇKA	0.99	0	1	0.99	0.99	0	1	0.98	0.99	0	1	0.99	0.96	0.02	0.99	0.92
DVM (polykernel)	0.96	0	0.98	0.95	0.97	0	0.98	0.96	0.97	0	0.98	0.96	0.92	0	0.96	0.89
DVM (Normal polykernel)	0.96	0	0.98	0.94	0.96	0	0.98	0.95	0.97	0	0.98	0.96	0.89	0	0.94	0.86
DVM (Puk)	0.98	0	0.99	0.97	0.97	0	0.98	0.96	0.98	0	0.99	0.97	0.99	0.04	0.97	0.95
DVM (RTFKernel)	0.92	0	0.96	0.89	0.92	0	0.96	0.90	0.92	0	0.96	0.89	0.82	0	0.91	0.77

Çizelge 3.3'te**,** sınıflandırma algoritmasının Doğru-Pozitif (DP), Yanlış-Pozitif (YP), Kappa ve ROC analizinin sonuçlarını gösterir. DP, sınıflandırma algoritması sonucunda aslında hasta olan ve hasta olmayan kişilerin oranını ifade eder. YP, sınıflandırma algoritmasının bir sonucu olarak gerçekten hasta olan ancak hasta olmadığı tespit edilen kişilerin oranına işaret eder. Bu, DP oranının 1'e ve YP oranının 0'a yakın olması gerektiği anlamına gelir.

Çizelge 3.3 incelendiğinde 3 yöntemin Roc analizinin 1 olarak hesaplanmış olduğu görülmektedir. Bu yöntemler ÇKA, Naive Bayes ve Rassal Orman. Yani, 3 yöntemin tümü en yüksek tanı oranına sahiptir. Bununla birlikte, ilgilenilen özellik seçim yöntemlerinde ÇKA, Karar Ağacı, DVM(Puk) ve Rassal Orman'nin DP oranının çok daha yüksek olduğu görülmektedir. Kappa değerleri yönünden incelendiğinde Rassal ormanın sınıflandırma sonucu beklenen ve gözlenen değerler arasında mükkemmel bir uyum gösterdiği görülmektedir.

Çizelge 3.4 Farklı Özellik seçimleri ve sınıflandırma algoritmaları için performans sonuçları

	Özellik Seçim Yok	**Filtre**	**Korelasyon**	**Tutarlılık**
Naive Bayes	%95.00	% 97.5	% 97.00	% 95.75
k-NN(k=2)	% 96.25	%99.75	% 98.50	% 98.25
Karar Ağacı	% 99.00	% 98.25	% 99.00	% 97.75
Rassal Orman	% 100	% 99.5	% 99.75	% 98.75
RTF Ağı	% 98.50	%99.25	% 98.50	% 98.25
ÇKA	% 99.75	% 99.5	% 99.75	% 96.50
DVM (PolyKernel)	% 97.75	% 98.25	% 98.25	% 95.00
DVM (Normalized Poly Kernel)	% 97.50	%97.75	% 98.25	% 93.50
DVM (Puk)	% 98.75	% 98.5	% 98.75	% 98.00
DVM (RTFKernel)	% 95.00	%95.50	% 95.00	% 89.00

Farklı özellik seçim yöntemleri için çalışmada yer verilen sınıflandırma metotlarının performanslarını gösteren çizelge 3.4 incelendiğinde doğru sınıflandırma oranı en yüksek olan (% 99.75) 3 durum korelasyon tabanlı özellik yöntemi ile rassal orman ve mlp

sınıflandırma yöntemiyle elde edilirken, filtre özellik seçim yöntemin de k-NN sınıflandırma yönteminden elde edilmiştir. Çizelge 3.4 incelendiğinde Tutarlılık özellik seçim yöntemi uygulanarak yapılan sınıflandırma yöntemlerinden elde edilen performansların diğer özellik seçim yöntemleri uygulanarak yapılan sınıflandırma yöntemlerinin performanslarından daha düşük olduğu dikkat çekici bir sonuç olmakla beraber %89'luk doğru sınıflandırma oranı ile en düşük performans tutarlılık özellik seçimi uygulanıp dvm (RTFKernel) sınıflandırma yönteminden elde edilmiştir.

5. TARTIŞMA ve SONUÇ

KBH tanısında geçmiş çalışmalar incelendiğinde genellikle tıbbi yöntemler başarılı olmuştur. Bu yöntemlere ek olarak, son yıllarda teknolojik gelişmelere de bağlı olarak bilgisayar destekli çeşitli algoritmalar yardımıyla doktorlara yardımcı olacak yeni bir sistem geliştirmek, hem bozukluğun hızlı teşhisine yardımcı olacak hem de doktorların iş yükünü azaltacak, böylece doktorların daha verimli çalışmasını sağlayacaktır.

Ayrıca, hastalığın bir doktor tarafından teşhis edilmesinden sonra, tanının bilgisayarlı sistemlerle doğrulanması de insan yapımı hataları ortadan kaldırmış olacaktır. Bu çalışmada 10 farklı sınıflandırma algoritması kullanılmıştır. Ayrıca DVM'nin 4 farklı çekirdek fonksiyonları kullanılmıştır. Çizelge 3.5'te 400 bireye ait aynı veri seti kullanılarak yapılan daha önceki çalışmalar ve kullandıkları sınıflandırma yöntemlerine ilişkin performans istatistikleri verilmektedir.

Çizelge 3.5 Kronik böbrek hastalığı veri seti için daha önce yapılmış çalışmalardan elde edilen doğruluk oranları

Çalışma	Program	Sınıflandırma Algoritması	Doğruluk
Celik vd.	WEKA	Karar Ağacı	%91.66
		DVM	%96.11
Charleonnan vd.	WEKA ve MATLAB	k-NN	%98.10
		DVM	%98.30
		Lojistik Regresyon	%96.55
		Karar Ağacı	%94.80
Chetty vd.	WEKA	Naive Bayes	%95.00
		DVM	%97.75
		k-NN	%95.75
		Karar Ağacı	%91.00
Gunarathne vd.	---	Karar Ormanları	%99.10
		Lojistik Regresyon	%95.00
		YSA	%97.50

Çizelge 3.5 incelendiğinde Çelik vd. (2016)'nin yapmış olduğu çalışmada Weka programı aracılığıyla sınıflandırma algoritmalarından Karar Ağacı %91.66 ve DVM'de %96.11 doğru sınıflandırma oranlarını elde etmişlerdir.

Charleonnan vd. (2016)'nin yapmış olduğu çalışmada Weka ve MATLAB programları aracılığıyla sınıflandırma algoritmalarından k-NN ile %98.10, DVM ile %98.30, Lojistik Regresyon ile %96.55 ve Karar Ağacı ile %94.80 doğru sınıflandırma oranlarını elde etmişlerdir.

Chetty vd. (2015)'i tarafından yapılan çalışmada Weka programı kullanılarak sınıflandırma algoritmalarından naive bayes ile %95.00, DVM ile %97.75, k- En Yakın Komşuluk ile %95.75 ve Karar Ağacı ile %91.00 doğru sınıflandırma oranlarını elde etmişlerdir.

Gunarathne vd. (2017)'i tarafından yapılan çalışmada sınıflandırma algoritmalarından Karar Ormanları ile %99.10, Lojisitk Regresyon ile %95.00 ve YSA ile %97.50 doğru sınıflandırma oranlarını elde etmişlerdir.

Çizelge 3.5 incelendiğinde daha önce yapılan çalışmalar arasında en yüksek performansın Gunarathne vd.'nin yaptıkları ve 99,10% ile Karar Ağacı sınıflandırma algoritması elde edilmiştrir. Daha önce yapılan çalışmaların tümünde KBH veri setindeki 25 değişkenin tamamı işleme dahil edilmiş olup çapraz geçerlilik(cross-validation) yapılmadan bu sonuçlar elde edilir iken yapılan bu çalışmadan elde edilen tüm sonuçlar çapraz geçerlilik sonucu elde edilmiştir.

Bu bağlamda bu çalışmanın diğer çalışmalardan daha iyi sonuç elde etmesi çalışmanın özgün sonuçlarından birisidir. Bu çalışmanın bir diğer özelliği ise diğer çalışmalarla kıyaslandığında farklı özellik seçimleri sonucunda değişken sayısında indirgemeye gitmiş ve daha az değişken ile daha önce yapılan birçok çalışmadan daha verimli sonuçlar elde edilmiştir.

6. KAYNAKLAR

Ababaei B, Sohrabi T, Mirzaei F, 2012, Assessment of radial basis and generalized regression neural networks in daily reservoir inflow simulation, Computer Science and Engineering, 42, 6074–6077.

Adak M F, Yurtay N, 2013, Algoritmasını Kullanarak Karar Ağacı Oluşturmayı Sağlayan Bir Yazılımın Geliştirilmesi, Bilişim Teknolojileri Dergisi, 6, 1–5.

Arı A, Berberler M E, 2017, Yapay Sinir Ağları ile Tahmin ve Sınıflandırma Problemlerinin Çözümü İçin Arayüz Tasarımı, Acta Infologica, 1, İstanbul.

Atılgan E, 2011, Karayollarında meydana gelen trafik kazalarının karar ağaçları ve birliktelik analizi ile incelenmesi, Hacettepe Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisan Tezi, 88s, Ankara.

Akçetin E, Çelik U, 2014, İstenmeyen Elektronik Posta (Spam) Tespitinde Karar Ağacı Algoritmalarının Performans Kıyaslaması, İnterner Uygulamaları ve Yönetimi Dergisi, 5, 43–56.

Akın C, Saraçlı S, 2014, Entropi Metoduyla Türk Lehçeleri Üzerinde Metinlerarası Bir Karşılaştırma, Journal of Turkish Studies, 9, 1–7.

Akman M, 2010, Veri Madenciliğine Genel Bakış ve Random Forests Yönteminin İncelenmesi: Sağlık Alanında Bir Uygulama, Ankara Üniversitesi, Sağlık Bilimleri Enstitüsü, Yüksek Lisans Tezi, 82s, Ankara.

Akpolat T, Utaş C, 2008, Hemodiyaliz Hekimi El Kitabı, Türk Nefroloji Derneği, Anadolu Yayıncılık.

Ay Ö, 2019, Özellik Seçimi Problemleri İçin Polihedral Konik Fonksiyonlar Temelli Çözüm Yaklaşımı. Eskişehir Teknik Üniversitesi, Endüstri Mühendisliği Anabilim Dalı, Yüksek Lisans Tezi, 43s, Eskişehir.

Aydın C, 2018, Makine Öğrenmesi Algoritmaları Kullanılarak İtfaiye İstasyonu İhtiyacının Sınıflandırılması, Avrupa Bilim ve Teknoloji Dergisi, 14, 169–175.

Ayhan S, Erdoğmuş Ş, 2014, Destek Vektör Makineleriyle Sınıflandırma Problemlerinin Çözümü İçin Çekirdek Fonksiyonu Seçimi, Eskişehir Osmangazi Üniversitesi, İktisadi ve İdari Bilimler Dergisi, 9, 175–198.

Azofra A A, Benitez J M, Castro J L, 2008, Consistency measures for feature selection, Journal of Intelligent Information, 30, 273–292.

Bekiryazıcı Ş, 2020, Elektroensefalografi İşaretlerinin Makine Öğrenmesi Algoritmaları İle İncelenmesi ve Sınıflandırılması, Bursa Uludağ Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 89s, Bursa.

Beyazıt B E, 2019, Büyük Veri Problemlerinde Performans Arttırmaya Yönelik Özellik Seçimi ve Boyut İndirgeme Optimizasyonu. Gazi Üniversitesi, Bilişim Enstitüsü, Yüksek Lisans Tezi, 63s, Ankara.

Bilişik M T, 2011, Destek Vektör Makinesi, Çoklu Regresyon Ve Doğrusal Olmayan Programlama İle Perakendecilik Sektöründe Gelir Yönetimi İçin Dinamik Fiyatlandırma, XI. Üretim Araştırmaları Sempozyumu, 23-24 Haziran, İstanbul 785–799.

Breiman L, 2001, Random Forests, Machine Learning, 45, 5–32.

Budak H, 2015, Özellik Seçim Yöntemleri ve Yeni Bir Yaklaşım, Mimar Sinan Güzel Sanatlar Üniversitesi, Fen Bilimleri Enstitüsü, Doktora Tezi, 125s, İstanbul.

Bulut F, 2016, Performance evaluations of supervised learners on imbalanced datasets, In Electric Electronics, Computer Science, Biomedical Engineerings Meeting (EBBT), IEEE, 1–4.

Bulut F, 2017, Bilgi Kuramındaki Entropi Kavramıyla İlgili Farklı Matematiksel Modeller, Bilge International Journal of Science and Technology Research, 1, 167–174.

Canedo V B, Maroño N S, Betansoz A A, Benítez J M, Herrera F, 2014, A review of microarray datasets and applied feature selection methods, Science Direct, Information Sciences, 282, 111–135.

Congalton R G, Green K, 1998, Assessing the Accuracy of Remotely Sensed Data: Principles and Practices, first edn. Lewis Publications, 137p, Boca Raton.

Charleonnan P, Fufaung T, Niyomwong T, Chokchueypattanakit W, Suwannawach S, vd., 2016, Predictive analytics for chronic kidney disease using machine learning techniques, The 2016 Management and Innovation Technology International Conference, Bangkok, Thailand, 81–83.

Chetty N, Vaisla K S, Sudarsan S D, 2015, Role of attributes selection in classification of chronic kidney disease patients, International Conference on Computing, Communication and Security (ICCCS), Le Meridien, Mauritius, 1–6.

Choubey D K, Paul S, Kumar S, 2017, Classification of Pima indian diabetes dataset using naive bayes with genetic algorithm as an attribute selection, In Communication and computing systems: proceedings of the international conference on communication and computing system, 451–455.

Cihan P, Kalıpsız O, 2015, Öğrenci Proje Anketlerini Sınıflandırmada En Başarılı Algoritmanın Belirlenmesi, Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi, 8, 41–49.

Cover T, Hart P, 1967, Nearest neighbor pattern classification, IEEE Trans. Inf. Theory 13, 21–27.

Çalış A, Kayapınar S, Çetinyokuş T, 2014, Veri Madenciliğinde Karar Ağacı Algoritmaları ile Bilgisayar ve İnternet Güvenliği Üzerine Bir Uygulama, Endüstri Mühendisliği Dergisi, 25, 2–19.

Çelik E, Atalay M, Kondiloğlu A, 2016, The diagnosis and estimate chronic kidney disease using the machine learning methods. International Journal of Intelligent Systems and Applications in Engineering. 4, 27–31p.

Çölkesen I, 2009, Uzaktan Algılamada İleri Sınıflandırma Tekniklerinin Karşılaştırılması ve Analizi, Gebze Yüksek Teknoloji Entitüsü, Jeodezi ve Fotogrametri Mühendisliği, Yüksek Lisans Tezi.

Çınar A, 2019, Veri Madenciliğinde Sınıflandırma Algoritmalarının Performans Değerlendirmesi Ve R Dili İle Bir Uygulama, Marmara Üniversitesi Öneri Dergisi, 14, 90–111.

Çifçi F, 2018, Öznitelik Seçme ve Makine Öğrenmesi Yöntemleriyle Eğitmen Performansının Tahmin Edilmesi, Anadolu Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 53s, Eskişehir.

Dash M, Liu H, 1997, Feature Selection for Classification, İntelligent Data Analysis, 131–156.

Dash M, Liu H, 2003, Consistency-based search in feature selection, ScienceDirect, 151, 155–176.

Demiraslan M, Suner A, 2021, Sağlık Veri Setlerinde Öznitelik Seçiminin Sınıflandırma Performansına Etkisi, Sağlık Bilimlerinde Yapay Zeka Dergisi, 1, 6–11.

Demirçalı A, 2015, Güç Transformatörü Hatalarının Destek Vektör Makineleri Yaklaşımıyla Belirlenmesi, Pamukkale Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi,78s, Denizli.

Dilki G, Başar Ö D, 2020, İşletmelerin İflas Tahmininde K- En Yakın Komşu Algoritması Üzerinden Uzaklık Ölçütlerinin Karşılaştırılması, İstanbul Ticaret Üniversitesi, Fen Bilimleri Dergisi, 19, 224–233.

Doak J, 1992, An Evaluation of Feature Selection Methods and Their Application to Computer Security, Computer Science, Technical report, Univ. of California at Davis, Dept.

Durgabai R P I, 2014, Feature Selection using ReliefF Algorithm, International Journal of Advanced Research in Computer and Communication Engineering, 3, 8215–8218.

Emel G G, Taşkın Ç, 2005, Veri Madenciliğinde Karar Ağaçları ve Bir Satış Analizi Uygulaması, Eskişehir Osmangazi Üniversitesi, Sosyal Bilimler Dergisi, 6, 221–229s.

Ezirmik A H, 2020, Meta-Sezgisel Yöntemler ile Müzik Verisi Üzerinde Özellik Seçimi ve Kategorizasyon, Eskişehir Osmangazi Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 61s, Eskişehir.

Filiz E, Karaboğa H A, Akoğul S, 2017, Bıst-50 Endeksi Değişim Değerlerinin Sınıflandırılmasında Makine Öğrenmesi Yöntemleri ve Yapay Sinir Ağları Kullanımı, Ç. Ü. Sosyal Bilimler Enstitüsü Dergisi, 26, 231–241.

Forman G, 2003, An Extensive Empirical Study of Feature Selection Metrics for Text Classification, Journal of Machine Learning Research, 3, 1289–1305.

Galit S, Nitin R P, C.B. Peter C B, 2010, Data Mining for Business Intelligence: Concepts, Techniques, and Applications in Microsoft Office Excel with Xlminer, Hoboken, Canada: Wiley.

Gazeloğlu C, 2020, Prediction of Heart Disease by Classifying with Feature Selection and Machine Learning Methods, Progress in Nutrition, 22, 660–670.

Girosi F, Pogio T, 1990, Networks and the best approximation property, Biological Cybernetics, 63, 169–176.

Gordon G, Pressman I, 1983, Quantitative Decision-Making For Business. Second Edition, Prentice Hall International Inc., USA

Gör İ, 2016, Çok Katmanlı Algılayıcı Yapay Sinir Ağı ile Lineer Diferansiyel Denklem Sisteminin Çözümü, 18. Akademik Bilişim Konferansı, 3-5 Şubat, Aydın.

Gunarathne W H S D, Perera K D M, Kahandawaarachchi, 2017, Performance evaluation on machine learning classification techniques for disease classification and forecasting through data analytics for chronic kidney disease (CKD). IEEE 17th International Conference on Bioinformatics and Bioengineering, 291–296p.

Guyon I, Weston J, Barnhil S, Vapnik V, 2002, Gene selection for cancer classification using support vector machines, Machine Learning, 46, 389–422.

Güldoğan E, 2017, Çeşitli Çekirdek Fonksiyonları İle Oluşturulan Destek Vektör Makinesi Modellerinin Performanslarının İncelenmesi: Bir Klinik Uygulama, İnönü Üniversitesi ve Mersin Üniversitesi, Biyoistatistik Ve Tıp Bilişimi Anabilim Dalı, Doktora Tezi, 88s, Malatya.

Günal S, 2008, Örüntü Tanıma Uygulamalarında Altuzay Analiziyle Öznitelik Seçimi ve Sınıflandırma, Eskişehir Osmangazi Üniversitesi, Fen Bilimleri Enstitüsü, Doktora Tezi, 97s, Eskişehir.

Hall M A, 1999, Corelation-based Feature Selection Machine Learning, The University of Waikato, Ph.D. Thesis, 178p, NewZealand.

Harrington P, 2012, Machine Learning in Action, Manning Publications Co.

Han J, Kamber M, Pei J, 2011, Data Mining Concepts and Techniques Third Edition (Third Edition), Morgan Kaufmann, Massachusetts.

Hastie T, Tibshirani R, Friedman J H, 2009, The elements of statistical learning: data mining, Inference and Prediction, 2nd edn. Springer, New York, USA, 533.

Ho T K, 1998, The Random Subspace Method for Constructing Decision Forests, IEEE Transactions on Pattern Analysis and Machine Intelligence, 20, 832–844.

Holzinger A, 2017, Introduction to machine learning knowledge extraction (MAKE), Machine Learning Knowledge Extraction, 1, 1–20.

Hu L Y, Huang M W, Ke S W, Tsai C F, 2016, The distance function effect on k-nearest neighbor classification for medical datasets, SpringerPlus, 5, 1–9.

İlkuçar M, 2015, Kronik Böbrek Hastalarının Yapay Sinir Ağı ve Radyal Temelli Fonksiyon Ağı ile Teşhisi, Mehmet Akif Ersoy Üniversitesi, Fen Bilimleri Enstitüsü Dergisi, 6, 82–88.

İşeri İ, Arıman S, 2019, Sedimandaki Ağır Metal Konsantrasyonunun Çoklu Değişken Regresyon Modelleri ve Çok Katmanlı Algılayıcı Ağ Modeli ile Tahmini, Avrupa Bilim ve Teknoloji Dergisi, 389–397.

Jiang L, Zhang H, Cai Z A, 2019, A novel bayes model: hidden naive bayes, IEEE T Know Data En, 21, 1361–1371.

Kagoda P A, Ndiritu J, Ntuli C, Mwaka B, 2010, Application of radial basis function neural networks to short-term streamflow forecasting, Physics and Chemistry of the Earth, 35, 571–581.

Karakaş M, 2020, Sınıflandırma Problemlerinde Özellik Seçimi için Karşıtlık Tabanlı Gri Kurt Optimizasyon Algoritması, Bilecik Şeyh Edebali Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 66s, Bilecik.

Karakoyun M, Hacıbeyoğlu M, 2014, Biyomedikal Veri Kümeleri ile Makine Öğrenmesi Sınıflandırma Algoritmalarının İstatistiksel Olarak Karşılaştırılması, Dokuz Eylül Üniversitesi, Mühendislik Bilimleri Dergisi, 16, 30–41.

Karegowda A G, Manjunath A S, Jayaram M A, 2010, Comparatıve Study of Attrıbute Selectıon Usıng Gaın Ratıo and Correlatıon Based Feature Selectıon, International Journal of Information Technology and Knowledge Management, 2, 271–277.

Kavzaoğlu T, Çölkesen İ, 2010, Karar Ağaçları ile Uydu Görüntülerinin Sınıflandırılması: Kocaeli Örneği, Harita Teknolojileri Elektronik Dergisi, 2, 36–45.

Kaya M, 2014, Gen İfade Verilerinde Öznitelik Seçimi ve Sınıflandırma, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisan Tezi, 82s, Ankara.

Kaynar O, Tuna M F, Görmez Y, Deveci M A, 2017, Makine Öğrenmesi Yöntemleriyle Müşteri Kaybı Analizi, Cumhuriyet Üniversitesi İktisadi ve İdari Bilimler Dergisi. 18, 1–14.

Ke´gl B, Krzyzak A, Niemann H, 2000, Radial basis function networks and complexity regularization in function learning and classification, Proceedings of the Fifteenth International Conference on Pattern Recognition, 2, 81–86.

Kesenek Y, 2019, Zararlı Yazılım Kaynaklı Veri Kaçırma Ataklarına Karşı Doküman Sınıflandırma Algoritması Geliştirme, Sakarya Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 92s, Sakarya.

Kılıç S, 2015, Kappa Test, Journal of Mood Disorders, 5, 142–144.

Kılınç D, Borandağ E, Yücalar F, Tunalı V, Şimşek M, Özçift A, 2016, k-NN Algoritması ve R Dili ile Metin Madenciliği Kullanılarak Bilimsel Makale Tasnifi, Marmara Fen Bilimleri Dergisi, 3, 89–94.

Kittler J, 1978, Feature set search algorithms. In: C.H. Chert, Ed., Pattern Recognition and Signal Processing, Sijthoff and Noordhoff, Mphen aan den Rijn, Netherlands, 41–60.

Kocatürk A, Turfan D, Altunkaynak B, 2019, Fılterıng Methods Used for Feature Selectıon ın Gene Expressıon Data, X. International Multidisciplinary Congress of Eurasia, 24-26 April, 67–75, Antalya.

Koç İ, 2016, Sınıflandırma Problemlerinde Meta-Sezgisel Optimizasyon Yöntemlerinin Özellik Seçimi ve Ayrıklaştırma Amacıyla Kullanımı, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 181s, Konya.

Koşan M A, Coşkun A, Karacan H, 2019, Yapay Zekâ Yöntemlerinde Entropi, Journal of Information Systems and Management Research, 1, 15–22.

Köktürk F, 2012, K-En Yakın Komsuluk, Yapay Sinir Ağları ve Karar Ağaçları Yöntemlerinin Sınıflandırma Basarılarının Karsılastırılması, Bülent Ecevit Üniversitesi, Sağlık Bilimleri Enstitüsü, Doktora Tezi, 77s, Zonguldak.

Kuzey C, 2012, Veri Madenciliğinde Destek Vektör Makinaları ve Karar Ağaçları Yöntemlerini Kullanarak Bilgi Çalışanlarının Kurum Performansı Üzerine Etkisinin Ölçülmesi ve Bir Uygulama, İstanbul Üniversitesi, Sosyal Bilimler Enstitüsü, Doktora Tezi, 318s, İstanbul.

Küçük H, Tepe C, Eminoğlu İ, 2013, Classification of EMG signals by k-Nearest Neighbor algorithm and Support vector machine methods, Signal Processing and Communications Applications Conference (SIU), 24-26 April, Haspolat.

Landis J R, Koch G G, 1977, The measurement of observer agreement for categorical data, Biometrics, 33, 159–174.

Lezki Ş, 2014, Çok Kriterli Karar Verme Problemlerinde Karar Ağacı Kullanımı, Siirt Üniversitesi, İktisadi Yenilik Dergisi, 2, 16–31.

Liu H, Setiono R, 1995, Chi2: Feature Selection and Discretization of Numeric, Proceedings of 7th IEEE International Conference on Tools with Artificial Intelligence, 388-391.

Liu H, Setiono R, 1996, A probabilistic approach to feature selection-a filter solution, Citeseer, 319–327.

Lubaib P, Ahammed V D, 2015, The heart defect analysis based on PCG signals using pattern recognition techniques, Procedia Technology, 24, 1024–1031.

Maillo J, Luengo J, Garc´ ıa S, Herrera F A, 2018, Preliminary Study On Hybrid Spill-Tree Fuzzy K-Nearest Neighbors for Big Data Classification, IEEE International Conference on Fuzzy Systems.

Marill T, Green D M, 1963, On the effectiveness of receptors in recognition system, IEEE Trans. Inform. Theory, 9, 11–17.

Marono N S, Beletansoz A A, Sanroman M T, 2007, Filter Methods for Feature Selection a Comparative Study, Intelligent Data Engineering and Automated Learning, 178–187.

Mather P M, 2005, Computer Processing of Remotely-Sensed Images.

Mitchell T, 1997, Machine Learning, McGraw Hill, New York.

Molina L C, Belanche L, Nebot A, 2002, Feature Selection Algorithms: A Survey and Experimental Evaluation, IEEE International Conference on Data Mining, Dec. 9-12, Maebashi City, Japan, 306–313.

Moradkhani H, Hsu K, Gupta H V, Sorooshian S, 2004, Improved streamflow forecasting using self-organizing radial basis function artificial neural networks, Journal of Hydrology, 295(1–4), 246–262.

Nasr M, El-Bahnasy K, Hamdy M, Kamal S M, 2017, A novel model based on non invasive methods for prediction of liver fibrosis. 13th International Conference on Electronics and Communication, 28-29 December, Pattaya, Tayland.

Novakovic J, Strbac P, Bulatovic D, 2011, Toward Optımal Feature Selectıon Usıng Rankıng Methods and Classıficatıon Algorıthms, Yugoslav Journal of Operations Research, 21, 119–135.

Okkan U, Dalkılıç H Y, 2012, Radyal Tabanlı Yapay Sinir Ağları ile Kemer Barajı Aylık Akımlarının Modellenmesi, İMO Teknik Dergi, 379, 5957–5966.

Omitaomu O A, 2006, Decision Trees. In Michael W. Berry and Murray Browne (Eds.),Lecture Notes in Data Mining, World Scientific Publishing of Hackensack, New Jersey, USA, 39–51.

Özkan Y, 2008, Veri Madenciliği Yöntemleri, Papatya Yayıncılık, İstanbul, 2008.

Pal M, 2005, Random forest classifier for remote sensing classification, International Journal of Remote Sensing, 26, 217–222.

Pal M, Mather P M, 2003, An assessment of the effectiveness of decision tree methods for land cover classification, Remote Sensing of Environment, 86, 554–565.

Pearson K, 1895, Contributions to mathematical theory of evolution: II. Skew variation in homogeneous material, Phil. Trans. Roy. Soc. London, 186, 343–414.

Polat K, 2008, Biyomedikal Sinyallerde Veri Ön-İşleme Tekniklerinin Medikal Teşhiste Sınıflama Doğruluğuna Etkisinin İncelenmesi, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Doktora Tezi, 151s, Konya.

Power M J D, 1987, Radial basis function for multivariable interpolation: a review"; In: Mason, J.C., Cox, M.G. (Eds.), Algorithms for Approximation. Clarendon Press, Oxford.

Pratama S F, Muda A K, Choo Y H, Muda N A, 2011, Computationally Inexpensive Sequential Forward Floating Selection for Acquiring Significant Features for Authorship Invarianceness in Writer Identification, International Journal of New Computer Architectures and their Applications, 1, 581–598.

Press W H, Teukolsky S A, Vetterling W T, Flannery B P, 1992, Numerical recipes in C (2nd ed.), the art of scientific computing, Cambridge University Press.

Pudil P, Novovicova J, Kittler J, 1994, Floating Search Methods in Feature Selection, Pattern Recognition Letters, 15, 1119–1125.

Quinlan J R, 1988, Decision Trees and Multivalued Attributes, In J. R. (ed.), Machine Intelligence, Ed. by J. Richards, Oxford Univ Press, Oxford, England, 11, 305–318.

Quinlan J R, 1993, C4.5: Programs for Machine Learning. Morgan Kaufmann, 302p, San Mateo, CA.

Rahman M M, Usman O L, Muniyandi R C, Sahran S, Mohomed S, Razak R A, 2020, Otizm Spektrum Bozukluğu için Özellik Seçim ve Sınıflandırmasına Yönelik Makine Öğrenim Yöntemlerinin Gözden Geçirilmesi, Brain Sciences Journal, 10, 1–26.

Shang W Q, Huang H K, Zhu H B, Lin Y M, Qu Y L, Wang Z H, 2007, A novel feature selection algorithm for text categorization, Expert Systems with Applications, 33, 1–5.

Shin K, Fernandes D, Miyazaki S, 2011, Consistency Measures for Feature Selection: A Formal Definition, Relative Sensitivity Comparison and a Fast Algorithm Proceedings of the 22nd International Joint Conference on Artificial Intelligence, 16-22 July, Barcelona, Catalonia, Spain, 1491–1497.

Silahtaroğlu G, 2013, Veri Madenciliği Kavram ve Algoritmaları, Papatya Yayincilik Eğitim, İstanbul.

Stearns S D, 1976, On selecting features for pattern classifiers, 3rd International Conference on Pattern Recognition, Coronado.

Suner A, Demiraslan M, 2021, Sağlık Veri Setlerinde Öznitelik Seçiminin Sınıflandırma Performansına Etkisi, Sağlık Bilimlerinde Yapay Zeka Dergisi, 1, 6–11.

Şahin E K, 2017, Özellik Seçimi Algoritmaları Kullanılarak Heyelanda Etkili Faktörlerin Belirlenmesi ve Heyelan Duyarlılık Haritalarının Üretilmesi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Doktora Tezi, 223s, İstanbul.

Şenol C, 2010, Yapay Sinir Ağı Ve Bulanık Mantık Hibrid Yapı Ve Algoritmalarının Geliştirilmesi, Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Doktora Tezi, 78s, İstanbul.

Takıcı H, 2018, Improvement of heart attack prediction by the feature selection methods, Turkish Journal of Electrical Engineering Computer Sciences, 26, 1–10.

Tapkan P, Özbakır L, Baykasoğlu A, 2011, Weka ile Veri Madenciliği Süreci ve Örnek Uygulama, Endüstri Mühendisliği Yazılımları ve Uygulamaları Kongresi, 30 Eylül-01/02 Ekim, İzmir, 247–262.

Taşcı E, Onan A, 2016, K-En Yakın Komşu Algoritması Parametrelerinin Sınıflandırma Performansı Üzerine Etkisinin İncelenmesi, Akademik Bilişim, 1–8, Aydın.

Tolun S, 2008, Destek Vektör Makineleri: Banka Başarısızlığının Tahmini Üzerine Bir Uygulama, İstanbul Üniversitesi, Sosyal Bilimler Enstitüsü, Doktara Tezi, 265s, İstanbul.

Topaloğlu M, 2014, Çevrimiçi Destek Vektör Makineleri Tabanlı Model Öngörülü Denetim, Pamukkale Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 112s, Denizli.

Turfan D, 2020, Gen Açıklama Verilerinin Sınıflandırılmasında Yeni Bir Özellik Seçimi Yöntemi, Hacettepe Üniversitesi, Fen Bilimleri Enstitüsü, Doktora Tezi, 67s, Ankara.

Uzer M S, 2014, Örüntü Tanıma Uygulamalarında Yapay Zekâ Ve Öznitelik Dönüşüm Metotları Kullanılarak Geliştirilen Öznitelik Seçme Algoritmaları, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Doktora Tezi, 103s, Konya.

Üstün B, Melssen W J, Buydens L M C, 2005, Facilitating the application of Support Vector Regression by using a universal Pearson VII function based kernel, Science Direct, Chemometrics and Intelligent Laboratory Systems 81, 29–40.

Vafeiadis T, Diamantaras K I, Sarigiannidis G, Chatzisavvas K C, 2015, A Comparison of Machine Learning Techniques for Customer Churn Prediction, Simulation Modelling Practice and Theory, 55, 1–9.

Vapnik V N, 1982, Estimation of Dependences Based on Empirical Data, Germany: Springer Verlag.

Vapnik V N, 1995, The nature of statistical learning theory, Springer, New York.

Vapnik V, 1998, Statistical Learning Theory, New York, John Willey.

Viera A J, Garrett J M, 2005, Understanding interobserver agreement: the kappa statistic. Family Medicine, 3, 360–370.

Wang G, Song Q, Sun H, Zhang X, 2013, A Feature Subset Selection Algorithm Automatic Recommendation Method, Journal of Artificial Intelligence Research, 47, 1–34.

Whitney A W, 1971, A direct method of nonparametric measurement selection, IEEE Trans. Comput, 20, 1100–1103.

Wyse N, Dubes R, Jain A K, 1980, A critical evaluation of intrinsic dimensionality algorithms, in: E.S. Gelsema and L.N. Kanal, (eds), Pattern Recognition in Practice, Morgan Kaufmann Publishers, Inc., 415–425.

Xing Y, Wang J, Zhao Z, 2007, Combination data mining methods with new medical data to predicting outcome of coronary heart disease, International Conference on Convergence Information Technology, Gyeongju, South Korea, 868–872.

Yazıcı B, Yaslı F, Gürleyik H Y, Turgut U O, Aktaş M S, Kalıpsız O, 2015, Veri Madenciliğinde Özellik Seçim Tekniklerinin Bankacılık Verisine Uygulanması Üzerine Araştırma ve Karşılaştırmalı Uygulama, 9. Ulusal Yazılım Mühendisliği Sempozyumu (UYMS-15), 15-17 Eylül 2015, İzmir, 72–83s.

Yıldız O, 2019, Derin öğrenme yöntemleriyle dermoskopi görüntülerinden melanom tespiti: Kapsamlı bir çalışma, Gazi Üniversitesi Mühendislik ve Mimarlık Fakültesi Dergisi, 34, 2241–2260.

Yıldız O, Tez M, Bilge H Ş, Akcayol M A, Güler İ, 2012, Meme Kanseri Sınıflandırması İçin Gen Seçimi, Gazi Üniversitesi Mühendislik ve Mimarlık Fakültesi Dergisi, 27, Ankara.

Yıldız A, Zan H, 2019, Segmantasyon yapmadan patolojik kalp sesi kayıtlarının tespiti için bir örüntü sınıflandırma algoritması, DÜMF Mühendislik Dergisi, 10, Diyarbakır.

Yoav F, Lıew M, 1999, The alternating decision tree learning algorithm, ICML ’99 Proceedings of the Sixteenth International Conference on Machine Learning, Bled, Slovenia, 124–133.

Zhang G, Ge H, 2013, Support vector machine with a Pearson VII function kernel for discriminating halophilic and non-halophilic proteins, Computational Biology and Chemistry, 46, 16–22.

Zhao Z, Liu H, 2007, Searching for interacting features, In Proceedings of the 20th International Joint Conference on Artifical Intelligence, Morgan Kaufmann Publishers Inc, 1156–1161.

İnternet Kaynakları

1– https://www.geeksforgeeks.org/ml-chi-square-test-for-feature-selection/, 11.06.2021

2– https://pypi.org/project/ReliefF/, 22.06.2021

3– https://tr.wikipedia.org/wiki/Fleiss%27in_kappa_katsayısı/, 22.06.2021

4– https://www.kisa.link/KU44/, 12.02.2021

5–https://blog.goldenhelix.com/cross-validation-for-genomic-prediction-in-svs/, 11.03.2021

6–https://www.openml.org/a/estimation-procedures/7, 11.03.2021

7–https://towardsdatascience.com/support-vector-machines-svm-c9ef22815589/, 09.04.2021

8–https://towardsdatascience.com/svm-classifier-and-rbf-kernel-how-to-make-better-models-in-python-73bb4914af5b/, 09.04.2021

9–https://waikato.github.io/wekawiki/getting_help/weka.attributeSelection.FilteredSubsetEval/, 05.11.2020

ÖZGEÇMİŞ

Adı Soyadı : Mustafa Demir

Doğum Yeri ve Tarihi : Artvin-11.02.1993

Yabancı Dili : İngilizce

İletişim (Telefon/E-posta) : mustafademir08@gmail.com

Eğitim Durumu (Kurum ve Yıl)

Lise : Zonguldak Atatürk Lisesi (2007-2011)

Lisans : Afyon Kocatepe Üniversitesi, İstatistik Böl. (2012-2017)

Yüksek Lisans : Afyon Kocatepe Üniversitesi, Fen Bilimleri Ens., İstatistik ABD, (2019-2021)

Çalıştığı Kurum/Kurumlar ve Yıl

: İstatistik, Yöneylem, Aktüerya Uygulama ve Araştırma Merkezi (2020-2021)