T.C.
İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**Doktora Tezi**

# MAKİNE ÖĞRENMESİ UYGULAMALARINDA ÖZNİTELİK SEÇİMİ ÜZERİNE YENİ BİR YAKLAŞIM

**Emrah SEZER**

**Enformatik Anabilim Dalı**

**Enformatik Programı**

**DANIŞMAN**
**Doç. Dr. Çiğdem EROL**

**Şubat, 2023**

**İSTANBUL**

Bu çalışma, 23.02.2023 tarihinde aşağıdaki jüri tarafından Enformatik Anabilim Dalı, Enformatik Programında Doktora tezi olarak kabul edilmiştir.

**Tez Jürisi**

Doç. Dr. Çiğdem EROL (Danışman)
İstanbul Üniversitesi
Fen Fakültesi

Doç. Dr. Murat GEZER
İstanbul Üniversitesi
Enformatik Bölümü

Doç. Dr. Özgür ÇAKIR
Marmara Üniversitesi
İşletme Fakültesi

Prof. Dr. Hakan YILDIRIM
Marmara Üniversitesi
İşletme Fakültesi

Dr. Öğr. Üyesi Tutku TUNCALI YAMAN
Marmara Üniversitesi
İşletme Fakültesi

**İntihal Programı Beyanı**

20.04.2016 tarihli Resmi Gazete'de yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince; Bu Lisansüstü teze, İstanbul Üniversitesi'nin abonesi olduğu intihal yazılım programı kullanılarak Fen Bilimleri Enstitüsü'nün belirlemiş olduğu ölçütlere uygun rapor alınmıştır.

**Proje Destekleri**




**Tezden Üretilmiş Yayınların Künye Bilgileri**

# ÖNSÖZ

Tez çalışması boyunca desteği, güveni ve sabrı için danışman hocam sevgili Doç.Dr.Çiğdem Erol'a,

Ders dönemimde ve tez izleme süreçlerimde desteklerini esirgemeyen sevgili hocam Prof.Dr.Sevinç Gülseçen'e,

Yüksek lisans tez çalışmalarımdan bugüne kadar olan kesintisiz desteği ve muhabbeti için sevgili Doç. Dr. Özgür Çakır'a,

Tez jürimde yanımda bulunan sevgili Prof. Dr. Hakan Yıldırım'a,

Kontrol süreçlerindeki desteği için Dr. Fidan Bayraktutan'a,

Tez çalışmalarımın başladığı günlerde aramıza katılan ve çalışmaları dikkatle takip eden Duru'ya,

Tüm çalışmalarımda en büyük destekçim Hülya'ya,

Bugüne kadar olan hayatımda benden hiçbir şeylerini esirgemeyen Sebile'ye ve Fevzi'ye,

Yıllarca aynı odayı paylaştığımız Furkan'a,

Sonsuz teşekkürlerimi iletmeyi bir borç bilirim.

Şubat 2023 Emrah SEZER

# İÇİNDEKİLER

**Sayfa No**

**ÖNSÖZ** ... **iv**
**İÇİNDEKİLER** ... **v**
**ŞEKİL LİSTESİ** ... **vii**
**TABLO LİSTESİ** ... **viii**
**SİMGE VE KISALTMA LİSTESİ** ... **x**
**ÖZET** ... **xi**
**SUMMARY** ... **xiii**
**1. GİRİŞ** ... **1**
1.1 ÇALIŞMANIN AMAÇ VE KAPSAMI ... 3
1.2 ÇALIŞMANIN ÖZGÜNLÜĞÜ ... 3
1.3 ÇALIŞMANIN SINIRLILIKLARI ... 4
**2. GENEL KISIMLAR** ... **6**
2.1 MAKİNE ÖĞRENMESİ ... 6
2.2 ÖZNİTELİK SEÇİMİ ... 15
2.2.1 Sonuçlarına Göre Öznitelik Seçimi ... 16
*2.2.1.1 Alt Küme Sonucu Döndüren Yaklaşımlar* ... *16*
*2.2.1.2 Öznitelik Sıralaması Döndüren Yaklaşımlar* ... *17*
2.2.2 Kullanılan Ölçülere Göre Öznitelik Seçim Yöntemleri ... 17
2.2.3 Öznitelik Seçiminde Arama Yöntemleri ... 19
2.2.4 Sarmalayıcı ve Filtreleyici Yaklaşımlar ... 20
*2.2.4.1 Sarmalayıcı Yaklaşımlar* ... *21*
*2.2.4.2 Filtreleyici Yaklaşımlar* ... *21*
2.2.5 Öznitelik Seçim Algoritmaları ... 21
*2.2.5.1 Korelasyon Bazlı Öznitelik Seçimi* ... *21*
*2.2.5.2 Tutarlılık Bazlı Öznitelik Seçimi* ... *23*
2.3 ÖZNİTELİK SEÇİMİ VE OPTİMİZASYON ... 23
2.3.1 Tek Amaçlı Optimizasyon ... 24
2.3.2 Çok Amaçlı Optimizasyon ... 24
*2.3.2.1 Pareto Optimal* ... *27*
*2.3.2.2 Baskın Olmayan Sıralı Genetik Algoritma (NSGA-II)* ... *28*

2.4 LİTERATÜR TARAMASI ..... 31
**3. MALZEME VE YÖNTEM ..... 39**
3.1 ÇOK AMAÇLI KORELASYON VE TUTARLILIK BAZLI ÖZNİTELİK SEÇİMİ ..... 39
3.2 VERİ KÜMELERİ ..... 42
3.3 KULLANILAN ARAÇLAR ..... 43
**4. BULGULAR ..... 46**
4.1 MERİT DEĞERLERİNİN DAĞILIM GRAFİKLERİ ..... 46
4.2 ORTALAMALARA GÖRE KARŞILAŞTIRMA ..... 56
4.3 EN YÜKSEK PERFORMANSA GÖRE KARŞILAŞTIRMA ..... 61
4.4 EN YÜKSEK PERFORMANSA GÖRE SIRALAMALAR ..... 66
4.5 F1 SKOR STANDART SAPMA KARŞILAŞTIRMALARI ..... 69
**5. TARTIŞMA VE SONUÇ ..... 70**
**KAYNAKLAR ..... 81**
**EKLER ..... 95**
EK-1. ÇOKOTUÖS ALGORİTMASI ..... 95
EK-2. VERİ KÜMESİNE GÖRE EN YÜKSEK DOĞRULUK SAĞLAYAN SINIFLANDIRICILAR ..... 102
EK-3. DETAYLI SINIFLANDIRMA SONUÇLARI ..... 103
**ÖZGEÇMİŞ ..... 123**

# ŞEKİL LİSTESİ

**Sayfa No**

**Şekil 2.1:** CRISP-DM adımları (Chapman ve diğ., 2000; Çakır, 2008)..... 7

**Şekil 2.2:** Model yaşam döngüsü adımları (Gunning ve diğ., 2019). ..... 9

**Şekil 2.3:** Öznitelik seçimine bakış açıları ..... 16

**Şekil 2.4:** Karar uzayından hedef uzayına tanımlanan bir durumun gösterimi (Feng ve diğ., 2022). ..... 26

**Şekil 2.5:** Pareto cepheler (Deb ve diğ., 2002). ..... 28

**Şekil 2.6:** NSGA-II Prosedürü (Deb ve diğ., 2002). ..... 30

**Şekil 3.1:** ÇOKOTUÖS yönteminin akış şeması. ..... 40

**Şekil 4.1:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi – Breast..... 47

**Şekil 4.2:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi - Primary Tumor. ..... 48

**Şekil 4.3:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi – Lymphography..... 49

**Şekil 4.4:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi – Mushroom..... 50

**Şekil 4.5:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi – Spect Heart. ..... 51

**Şekil 4.6:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi – Chess. ..... 52

**Şekil 4.7:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi – Splice..... 53

**Şekil 4.8:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi – Audiology. ..... 54

**Şekil 4.9:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi - Voting Records. ..... 55

**Şekil 4.10:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi – Soybean. ..... 56

# TABLO LİSTESİ

**Sayfa No**

**Tablo 2.1:** Web of Science literatür taraması sorguları. ..... 31

**Tablo 3.1:** Tez çalışmasında kullanılan veri kümeleri. ..... 43

**Tablo 3.2:** Tez Çalışmasında kullanılan sınıflandırıcılar. ..... 45

**Tablo 4.1:** Ortalama karşılaştırması - Audiology veri kümesi. ..... 57

**Tablo 4.2:** Ortalama karşılaştırması - Breast veri kümesi. ..... 57

**Tablo 4.3:** Ortalama karşılaştırması - Chess veri kümesi. ..... 58

**Tablo 4.4:** Ortalama karşılaştırması - Lymphography veri kümesi. ..... 58

**Tablo 4.5:** Ortalama karşılaştırması - Mushroom veri kümesi. ..... 59

**Tablo 4.6:** Ortalama karşılaştırması - Primary Tumor veri kümesi. ..... 59

**Tablo 4.7:** Ortalama karşılaştırması - Soybean veri kümesi. ..... 60

**Tablo 4.8:** Ortalama karşılaştırması - Spect Heart veri kümesi. ..... 60

**Tablo 4.9:** Ortalama karşılaştırması - Splice veri kümesi. ..... 60

**Tablo 4.10:** Ortalama karşılaştırması - Voting Records veri kümesi. ..... 61

**Tablo 4.11:** En yüksek doğruluk karşılaştırması - Audiology veri kümesi. ..... 62

**Tablo 4.12:** En yüksek doğruluk karşılaştırması - Breast veri kümesi. ..... 62

**Tablo 4.13**: En yüksek doğruluk karşılaştırması - Chess veri kümesi. ..... 62

**Tablo 4.14:** En yüksek doğruluk karşılaştırması - Lymphography veri kümesi. ..... 63

**Tablo 4.15:** En yüksek doğruluk karşılaştırması - Mushroom veri kümesi. ..... 63

**Tablo 4.16:** En yüksek doğruluk karşılaştırması - Primary Tumor veri kümesi. ..... 64

**Tablo 4.17:** En yüksek doğruluk karşılaştırması - Soybean veri kümesi. ..... 64

**Tablo 4.18:** En yüksek doğruluk karşılaştırması - Spect Heart veri kümesi. ..... 64

**Tablo 4.19:** En yüksek doğruluk karşılaştırması - Splice veri kümesi. ..... 65

**Tablo 4.20:** En yüksek doğruluk karşılaştırması - Voting Records veri kümesi. ..... 65

**Tablo 4.21:** En yüksek doğruluk sıralamaları. ..... 66

**Tablo 4.22:** En yüksek sınıflandırma doğruluğu sırasına göre sıralamalar..... 66

**Tablo 4.23:** En iyi ve en kötü sıralama elde etme adetleri..... 67

**Tablo 4.24:**.Veri kümesine göre diğer yaklaşımları domine etme sayıları. ..... 67

**Tablo 4.25:** F1 Skor standart sapmalarına göre karşılaştırmalar..... 69

# SİMGE VE KISALTMA LİSTESİ

| Kısaltmalar | Açıklama |
|---|---|
| **BGA** | **:** İkili genetik algoritma |
| **BGWOPSO** | **:** Hibrit gri kurt optimizasyonu kullanarak ikili optimizasyon algoritması |
| **BMOChOA** | **:** İkili çok amaçlı şempanze optimizasyonu |
| **BPSO** | **:** İkili parçacık sürü optimizasyonu |
| **CFS** | **:** Korelasyon bazlı öznitelik seçimi |
| **ÇOKOTUÖS** | **:** Çok amaçlı korelasyon ve tutarlılık bazlı öznitelik seçimi |
| **CONS** | **:** Tutarlılık bazlı öznitelik seçimi |
| **EBMOChOA** | **:** İkili çok amaçlı hibrit filtre-sarmalayıcı şempanze optimizasyonu |
| **FS** | **:** Öznitelik Seçimi |
| **GWO** | **:** Hibrit gri kurt optimizasyonu |
| **HNB** | **:** Gizli naive bayes |
| **KNN** | **:** K – en yakın komşuluk algoritması |
| **MLOps** | **:** Makine öğrenmesi operasyonları |
| **MLP** | **:** Çok katmanlı algılayıcı |
| **MOCOCOFS** | **:** Çok amaçlı korelasyon ve tutarlılık bazlı öznitelik seçimi |
| **MOO** | **:** Çok amaçlı optimizasyon |
| **ÖS** | **:** Öznitelik seçimi |
| **PSO** | **:** Parçacık sürü optimizasyonu |
| **SCMFS** | **:** Paylaşılan ortak mod dikkate alınarak öznitelik seçimi |
| **SFS** | **:** Stokastik fraktal arama |
| **SVM** | **:** Destek vektör makinesi |
| **WOASAT-2** | **:** Öznitelik seçimi için benzetilmiş tavlama ile hibrit balina optimizasyon algoritması |

# ÖZET

## DOKTORA TEZİ

### MAKİNE ÖĞRENMESİ UYGULAMALARINDA ÖZNİTELİK SEÇİMİ ÜZERİNE YENİ BİR YAKLAŞIM

**Emrah SEZER**

**İstanbul Üniversitesi**
**Fen Bilimleri Enstitüsü**
**Enformatik Anabilim Dalı**

**Danışman : Doç. Dr. Çiğdem EROL**

Sınıflandırma çalışmalarında öznitelik seçimi uygulamaları ile sonuçların iyileştirilmesi sağlanmaktadır. Bu tezde, sınıflandırma için öznitelik seçiminde kullanılabilecek yüksek performanslı alternatif bir yaklaşım önerilmesi amaçlanmaktadır. Son yıllarda, çok amaçlı optimizasyon algoritmalarının öznitelik seçimi çalışmalarında kullanılması ile yüksek performanslı yaklaşımlar önerilmiştir. Bu tez çalışmasında, öznitelik alt kümesi değerlendirme yaklaşımına dayalı olarak hesaplanan korelasyon ve tutarlılık merit değerlerinin oluşturduğu kartezyen uzay üzerinde merit değerlerinin her ikisini de aynı anda maksimize edecek şekilde ideal öznitelik alt kümesi adaylarının belirlenmesi ve bu sınırlı sayıdaki aday için sarmalayıcı yaklaşım kullanılarak nihai öznitelik alt kümesinin seçilmesi önerilmektedir. Literatürde yapılan araştırmalara göre bu tez çalışmasında önerilen kartezyen uzay üzerinde daha önce çözüm arayışına rastlanmamıştır. Korelasyon ve tutarlılık merit değerlerinden oluşan kartezyen uzayda, Pareto ve NSGA-II kullanılarak, iki merit değerini de aynı anda maksimize eden adayların filtrelenmesi sağlanmıştır. Filtrelenen bu adaylar arasında karar vermek için ise her birine sarmalayıcı yaklaşım uygulanarak en yüksek sınıflandırma doğruluğu değerini sağlayan adayın seçilmesi önerilmiştir. Korelasyon ve tutarlılık metrikleri hesaplanması için üzerinde çalışılan veri kümesinde bulunan tüm özniteliklerin kategorik olması sağlanmak zorundadır. Bu nedenle, kategorik veri kümeleri üzerinde yüksek performans ile çalışmakta olan Catboost algoritması sarmalayıcı sınıflandırıcı olarak önerilmiştir. Sınıflandırma görevlerine yönelik çok amaçlı öznitelik alt kümesi seçim çalışmaları kapsamında araştırmacıların çoğunlukla tahmin hata oranını ve öznitelik sayısını minimize etme üzerine yoğunlaştıkları görülmektedir. Bu tez çalışması, iki farklı filtreleyici alt küme skorlama yaklaşımı ve optimizasyon algoritmaları yardımı ile oldukça az sayıda adayın sarmalayıcı aşamada değerlendirilmesini sağlamaktadır. Bu sayede, filtreleyici yaklaşımların sınıflandırma doğruluğu yüksekliğini garanti etmemesi ve sarmalayıcı yaklaşımların yüksek maliyet oluşturması risklerine karşı güçlü bir yaklaşım geliştirilmiştir. UCI veri deposundan temin edilen on farklı veri kümesi üzerinde, korelasyon ve tutarlılık bazlı öznitelik seçimi uygulamalarının yanısıra bu tezde önerilen ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGA II yaklaşımları uygulanmıştır. Uygulamalarda bu tezde önerilen iki yaklaşımın, üzerinde çalışılan on veri kümesinden yedi tanesinde en iyi sonucu sağlamasının

yanı sıra Primary Tumor, Breast ve Spect Heart veri kümelerinde korelasyon ve tutarlılık bazlı öznitelik seçimi uygulamalarına üstünlük sağladığı görülmüştür.
Şubat 2023, 137 sayfa.
**Anahtar kelimeler:** Öznitelik Seçimi, Çok Amaçlı Öznitelik Seçimi, Sınıflandırma, Makine Öğrenmesi

# SUMMARY


**A NEW APPROACH ON FEATURE SELECTION IN MACHINE LEARNING APPLICATIONS**

**Ph.D. THESIS**

**Emrah SEZER**

**İstanbul University**
**Institute of Graduate Studies in Sciences**
**Department of Informatics**

**Supervisor : Assoc. Prof. Dr. Çiğdem EROL**

In classification studies, the results are improved by feature selection applications. In this thesis, it is aimed to propose a high-performance alternative approach to be used in feature selection for classification. In recent years, high-performance approaches have been proposed using multi-objective optimization algorithms in feature selection studies. In this thesis, it is suggested that the ideal feature subset candidates are determined in a way that maximizes both merit values at the same time on the cartesian space formed by the correlation and consistency merit values calculated based on the feature subset evaluation approach and the final feature subset is selected for this limited number of candidates by using the wrapper approach. According to the literature research, a solution has yet to be found on the cartesian space proposed in this study. In the cartesian space consisting of correlation and consistency merit values, by using Pareto and NSGA-II, candidates who maximize both merit values at the same time are filtered. In order to decide among these filtered candidates, it is suggested to choose the candidate that provides the highest classification accuracy by applying a wrapper approach to each of them. In order to calculate correlation and consistency metrics, all attributes in the studied data set must be categorical. For this reason, the Catboost algorithm, which works on high-performance categorical datasets, is proposed as a wrapper classifier. Within the scope of multi-objective feature subset selection studies for classification tasks, researchers mainly focus on minimizing the estimation error rate and the number of features. This thesis study evaluates a few candidates in the wrapper stage with the help of two different filtering subset scoring approaches and optimization algorithms. In this way, a robust approach has been developed against the risks that the filtering approaches do not guarantee high classification accuracy and the high cost of wrapper approaches. On ten different datasets provided on the UCI platform, in addition to the correlation and consistency-based feature selection applications, the MOCOCOFS-Pareto and MOCOCOFS-NSGA II approach proposed in this thesis was applied. In practice, it has been observed that the two approaches proposed in this thesis provide the best

results in seven of the ten datasets studied, as well as superior to correlation and consistency-based feature selection applications in Primary Tumor, Breast, and Spect Heart datasets.
February 2023, 137 pages.
**Keywords:** Feature Selection, Multi-Objective Feature Selection, Classification, Machine Learning

# 1. GİRİŞ

Günümüz dünyasında enformasyon obezitesinin giderek büyüyen ve yaygınlaşan bir problem olması, veri bilimi çalışmalarına da olumsuz şekilde yansımaktadır. Sınıflandırma çalışmalarının veri toplama sürecindeki araştırmacılar, daha başarılı modeller elde etmek ve alışılagelmişin dışında örüntüler (*pattern*) tespit etme hedeflerine sahiptirler. Bu hedefler, araştırmacıların oldukça açgözlü şekilde öğrenme verisini genişletme eğilimine sahip olmasına neden olmaktadır. Bu eğilim, öğrenme verisi içerisine açıklayıcılığı düşük, ilgisiz (*irrelavant*) ve alakasız (*redundant*) özniteliklerin de dahil edilmesine neden olmaktadır. Bu durum öğrenme verisinin barındırdığı gürültünün artmasına neden olmaktadır. Sınıflandırma algoritmalarının da açgözlü davranarak tüm öğrenme verisinden faydalanma eğiliminde olması, verinin barındırdığı gürültüden olumsuz etkilenmekte ve tahmin modelinin başarısının düşmesine neden olmaktadır. Araştırmacıların ve sınıflandırma algoritmalarının aç gözlülüklerinin sınıflandırma çalışmalarına olan olumsuz etkilerini bertaraf etmek için ilgisiz ve alakasız özniteliklerin çalışma dışında bırakılmasına yönelik çalışmalara, öznitelik seçimi adı verilmektedir. Öznitelik seçimi çalışmaları kullanılarak geleneksel olarak;

(i) İşlem süresinin düşürülmesi,
(ii) Sınıflandırma başarısının arttırılması,
(iii) Daha basit modeller elde edilmesi

hedeflerine ulaşmak amaçlanmaktadır (Liu ve Motoda, 1998b). Bu hedeflere ek olarak günümüz şartlarında;

(iv) Bulut ortamlardaki kaynak kullanımına bağlı tüketim bedellerinin düşürülmesi,
(v) Enerji tüketiminin düşürülmesi

gibi güncel gereksinimlerden de bahsedilebilir. Bu hedeflerin hepsine aynı anda ulaşılamayabilir. Buna rağmen, araştırmacının kendi belirlediği önceliklere uygunluk gösteriyorsa çalışma başarılı kabul edilebilir.

İnsanların karar verme süreçleri ile algoritmaların karar verme süreçleri paralellik arz etmektedir. Yapay zeka veya makine öğrenmesi çalışmaları doğası gereği, çalışmanın

sonuçlarının başarı düzeyi öngörülemez bir durumda başlatılmaktadır ve oldukça yüksek efor gerektiren bu çalışmaların kolaylıkla fark edilebilen kalıplardan (*pattern*) daha ileri seviyede çıktılar üretmesi beklenmektedir. Teknolojinin ve algoritmaların çok daha sade olduğu, bu çalışmaların "*knowledge discovery from databases*" olarak adlandırıldığı dönemlerde bile bu prensibin var olduğu görülmektedir.

Öznitelik seçimi yaklaşımları çıktıları açısından, öznitelik alt kümesi seçimi ve öznitelik sıralama olmak üzere iki başlık altında incelenmektedir. Bu çerçevede, öznitelik alt kümesi seçim yaklaşımlarının öznitelik sıralama yaklaşımlarına göre nispeten daha çok kaynak tüketimine ve çalışma süresine ihtiyaç duyabildiği söylenebilir. Makine öğrenmesi literatüründeki gelişmeleri bu gözle değerlendirmek gerekirse, Quinlan tarafından ortaya atılan C4.5 karar ağaçları çıktı olarak bir ağaç şeklinde okunabilir bir diyagram vermektedir (Quinlan, 1987) ve bu çıktı bir açıklanabilir yapay zeka (*explainable artificial intelligence*) çalışması olarak kabul edilmektedir. Quinlan'ın C4.5 karar ağacını ortaya atmasından yıllar sonra, Ho'nun (1995) yaklaşımı ve Breiman (2001) tarafından bugün kullanılan şekli ile ortaya atılan rastgele orman (*random forest*) yaklaşımı geliştirilmiştir. Rastgele orman yaklaşımı, birçok ağaçtan oluşan bir model çıktısı verdiği için Quinlan (1987) tarafından geliştirilen yaklaşımdaki ağaç diyagramı çıktısını verebilme yeteneğini kaybetmiştir. Ayrıca, daha çok ağaç ürettiği için gereksinim duyduğu süre ve kaynak tüketimi de daha fazladır. Buna rağmen, rastgele orman algoritması birçok veri kümesi üzerinde daha yüksek tahmin doğruluğu (*predictive accuracy*) elde edebildiği görüldüğü için C4.5 algoritmasına iyi bir alternatif olmuştur. Benzer şekilde, öznitelik alt kümesi skorlama yaklaşımının avantajlarının ve çıktı kalitesinin, öznitelik sıralama (*feature ranking*) yaklaşımlarından daha isabetli sonuçlar üretmeye nispeten daha uygun olduğu ortaya konulabilirse araştırmacılar için makul bir seçenek olduğu söylenebilir. Bu tez çalışmasında, literatürde bulunan öznitelik alt kümesi seçim yaklaşımlarına alternatif daha iyi bir alt küme seçim yaklaşımı oluşturmaya çalışıldığı için sıralama yaklaşımları ile olan karşılaştırmalar kapsam dışı bırakılmıştır.

Bu tez çalışmasında, literatürde var olan korelasyon bazlı öznitelik alt kümesi seçimi *(correlation based feature subset selection - CFS)* ve tutarlılık bazlı öznitelik alt kümesi seçimi *(consistency based feature subset selection - CONS)* yaklaşımları ile her aday öznitelik alt kümesinin skorlanması sağlanmaktadır. Bu ikili skorlamanın ardından, iki skoru da maksimize etme hedefine yönelik olarak, elde edilen skorların Pareto optimal kümesi ve NSGA-II tarafından alternatif çözümler tespit edilmektedir. Pareto küme ve NSGA-II tarafından

belirlenen öznitelik alt kümeleri için, Catboost sınıflandırma algoritması uygulanan ve sınıflandırma doğruluğu değerine göre en yüksek değeri döndüren öznitelik alt kümesi seçimi yaklaşımının çıktısı olarak seçilmiş olmaktadır. Önerilen yaklaşımların döndürdüğü öznitelik alt kümeleri, CFS ve CONS ile seçilen alt kümeler ile 12 farklı sınıflandırma algoritmasına tabi tutulmuştur ve sonuçları karşılaştırmalı olarak incelenmiştir.

Bu tez çalışmasının genel yapısı şu şekildedir. Genel kısımlar bölümünde, çalışmada gerçekleştirilen uygulamaları açıklamaya yönelik olarak makine öğrenmesi ve alt başlıkları, öznitelik seçimi yaklaşımları, tek ve çok amaçlı optimizasyon uygulamaları hakkında bilgi verilmiştir. Ayrıca bu bölümde, çalışma ile ilgili detaylı bir literatür taraması da bulunmaktadır.

Malzeme ve yöntem bölümünde, çalışma kapsamında önerilmekte olan çok amaçlı korelasyon ve tutarlılık bazlı öznitelik alt kümesi seçimi (ÇOKOTUÖS) yaklaşımının işleyişi hakkında detaylı bir anlatım ve bu yaklaşımın iki farklı optimizasyon algoritması ile olan varyasyonlarının detaylı açıklamalarına yer verilmiştir. Ayrıca çalışmada kullanılan veri kümeleri ve sınıflandırıcılar hakkında bilgi verilmiştir.

Bulgular bölümünde, önerilen öznitelik alt kümesi seçimi yöntemlerinin 10 farklı veri kümesi üzerinde gerçekleştirilen uygulamalarına dair sonuçlar yer almaktadır.

Tartışma ve sonuçlar bölümünde, işbu tez çalışması kapsamında önerilmekte olan sınıflandırma görevlerine yönelik öznitelik alt kümesi seçimi yaklaşımlarının, çalışma içerisinde uygulanmış olan farklı optimizasyon yaklaşımları ile olan varyasyonlarının ve literatürde daha önceden bulunan yaklaşımların performanslarının karşılaştırılması bulunmaktadır. Ayrıca bu bölümde, gelecek çalışmalar hakkında bilgi verilmektedir.

## 1.1 ÇALIŞMANIN AMAÇ VE KAPSAMI

Bu tez çalışmasında, literatürde bulunan korelasyon ve tutarlılık bazlı öznitelik alt kümesi seçim yaklaşımlarının aynı anda kullanılarak daha yüksek sınıflandırma doğruluğu sağlayabilecek hibrit bir yaklaşımın ortaya konulması amaçlanmıştır.

## 1.2 ÇALIŞMANIN ÖZGÜNLÜĞÜ

Literatürde daha önce korelasyon ve tutarlılık bazlı öznitelik alt kümesi değerlendirmesine dayalı bir öznitelik alt kümesi seçim çalışması bulunmamaktadır. Korelasyon ve tutarlılık merit

değerlerinin oluşturduğu kartezyen düzlem üzerinde bu iki merit değerini de aynı zamanda maksimize etmeyi amaçlayan yaklaşım literatürde ilk defa önerilmektedir.

Çok amaçlı optimizasyon algoritmalarının seçtiği adayların teke indirgenmesi aşamasında sarmalayıcı yaklaşımın kullanılması ile, adaylar arasından yüksek sınıflandırma doğruluğu sağlayanının seçilmesine olanak sağlanmıştır. Bu aşama, sarmalayıcı yaklaşımların yüksek sınıflandırma başarısı elde etme fonksiyonundan faydalanılmasını sağlamaktadır.

Korelasyon ve tutarlılık bazlı öznitelik seçimi yaklaşımlarının uygulanacağı veri kümesinde sürekli öznitelikler varsa çalışma öncesi kesikleştirme uygulaması yapılmaktadır. Bu çalışma kapsamında, merit değerlerin hesaplanması için hazırlanmış olan tamamı kategorik özniteliklerden oluşan bu veri kümesinin, sarmalayıcı yaklaşımla seçilmek üzere değerlendirilen her öznitelik alt kümesi için CatBoost sınıflandırma algoritması kullanılmıştır. CatBoost algoritması, kategorik veri kümeleri üzerinde yüksek başarı elde etmek üzere Yandex AR-GE ekibi tarafından tasarlanmıştır (Prokhorenkova ve diğ., 2018).

Tamamı kategorik halde bulunan bir veri kümesi üzerinde yapılan tutarlılık hesaplaması, tüm kategorik özniteliklerin kukla (*dummy*) özniteliklere dönüştürülmüş hali ile yapılan K – en yakın komşuluk (KNN) hesaplaması ile birebir örtüştüğü, bu tez çalışması kapsamında yapılan analizlerde farkedilmiş olup, literatürde bu bilgiye rastlanılmamıştır.

## 1.3 ÇALIŞMANIN SINIRLILIKLARI

Bu tez çalışması kapsamında üzerinde çalışılan korelasyon ve tutarlılık kartezyen uzayı üzerinde daha önce çalışılmamış olmasına yönelik olarak detaylı araştırma yapılması ihtiyacı duyulmuştur. Bu çerçevede; Web of Science, Scopus, Google Scholar ve Ulusal Tez Merkezi üzerinden Tablo 2.1'de yer alan sorgular kullanılarak detaylı literatür taramaları yapılmıştır.

Bu tez çalışmasının kodlamasının yapıldığı Python 3.8 yazılım dilinde, korelasyon ve tutarlılık merit değerlerini hesaplamaya yönelik olarak güvenilir kütüphanelerde gerekli fonksiyonlar bulunmamaktadır. Bu sorun için, korelasyon bazlı merit hesaplaması için Github üzerinden kod desteği bulunmuştur ancak tutarlılık hesaplaması bu tez çalışması kapsamında özel olarak yazılmıştır.

Çok amaçlı optimizasyonun, Python 3.8 yazılım dilinde makine öğrenmesi çalışmaları için kullanıldığı kütüphane ve kod örneklerine ulaşılamamıştır. Öncelikle Pymoo kütüphanesinde

(Blank ve Deb, 2020) çalışmalar denenmiş ancak Pandas veri çerçevesi (*dataframe*) üzerindeki kolonları adresleyecek şekilde kesikli fonksiyonların tanımlanması bu kütüphane üzerindeki yapı ile mümkün olamamıştır. Akabinde JmetalPy kütüphanesi (Benítez-Hidalgo ve diğ., 2019) üzerinde çalışmalar devam ettirilmiş, kütüphanenin yayıncıları tarafından kesikli problemler için hazırlanmış olan kod örnekleri incelenmiş, bu örneklerden Pandas veri çerçevesi üzerinde bulunan değerleri adresleyecek şekilde amaç fonksiyonları tanımlanabilmiştir. Bu sayede NSGA-II yaklaşımının çalışma kapsamında uygulaması sağlanmıştır.

# 2. GENEL KISIMLAR

Bu bölümde, tez çalışması kapsamında kullanılan makine öğrenmesi, sınıflandırma, sınıflandırma için öznitelik seçimi ve optimizasyon yaklaşımları hakkında detaylı bilgi verilmiştir.

## 2.1 MAKİNE ÖĞRENMESİ

Makine öğrenmesi, yapay zeka araştırmalarının bir alt dalı olup günümüzde hemen her alanda kullanabileceğimiz çok sayıda yöntem sunmaktadır. Eğitim (Asthana ve Hazela, 2020), sağlık (Litjens ve diğ., 2017; Emre ve diğ., 2021), politika (Brennan ve Oliver, 2013), denetim (Dickey ve diğ., 2019) gibi birçok disiplinde çalışan alan uzmanlarının ilgisini çeken makine öğrenmesi yaklaşımları, veri kümelerinin analiz edilmesi ve bu analiz sonuçlarının yorumlanarak geleceğe dönük çıktılar elde edilmesine yardımcı olur.

Makine öğrenmesi kavramının literatürde çok sayıda tanımı yapılmıştır. Bu tanımlardan bazıları aşağıda yer almaktadır:

Mitchell (1997) makine öğrenmesini, bir bilgisayar programının G görevlerinde P ile ölçülen performansı, D deneyimi ile artıyorsa o bilgisayar programının bazı G görevlerinin sınıflarına ve P performans ölçüsüne göre D deneyiminden öğrendiği söylenmektedir şeklinde tanımlamıştır.

Flach (2012) makine öğrenmesinin doğru görevleri gerçekleştiren doğru modelleri oluşturmak için doğru özellikleri kullanmakla ilgili olduğunu ifade etmiştir.

Mohri ve diğ. (2018), makine öğrenmesini performansı iyileştirmek veya doğru tahminler yapmak için deneyim kullanarak hesaplama yöntemleri şeklinde tanımlamıştır.

Daha önceki yıllarda, veri tabanlarından bilgi keşfi (*knowledge discovery from databases*) ve veri madenciliği gibi başlıklar altında incelenen yaklaşımlar günümüzdeki en gelişmiş halleri ile makine öğrenmesi altında incelenmektedir. Bir bilgisayar, bir görevi, deneyim arttıkça daha yüksek performans ile yerine getiriyorsa, bu bilgisayarın bu deneyimden öğrendiği söylenebilir (Geurts ve diğ., 2006).

1996 yılında Daimler-Benz, SPSS ve NCR tarafından kurulan konsorsiyum, makine öğrenmesi çalışmalarının o dönemdeki adı olan veri madenciliği çalışmaları için altı temel aşamadan oluşan CRISP-DM adımlarının izlenmesini önermiştir. CRISP-DM adımları;

1- İşi anlama

2- Verileri anlama

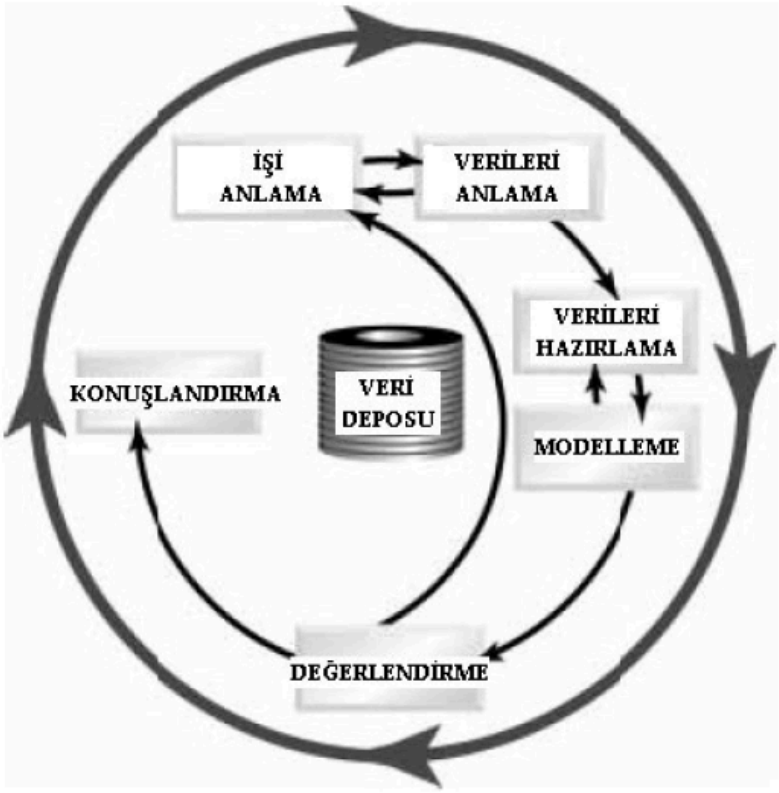
3- Verileri hazırlama

4- Modelleme

5- Değerlendirme

6- Kullanıcılarla Paylaşma (Konuşlandırma)

şeklinde tanımlanmıştır (Chapman ve diğ., 2000; Akpınar, 2014). CRISP-DM adımları Şekil 2.1'deki gibi görselleştirilmiştir.

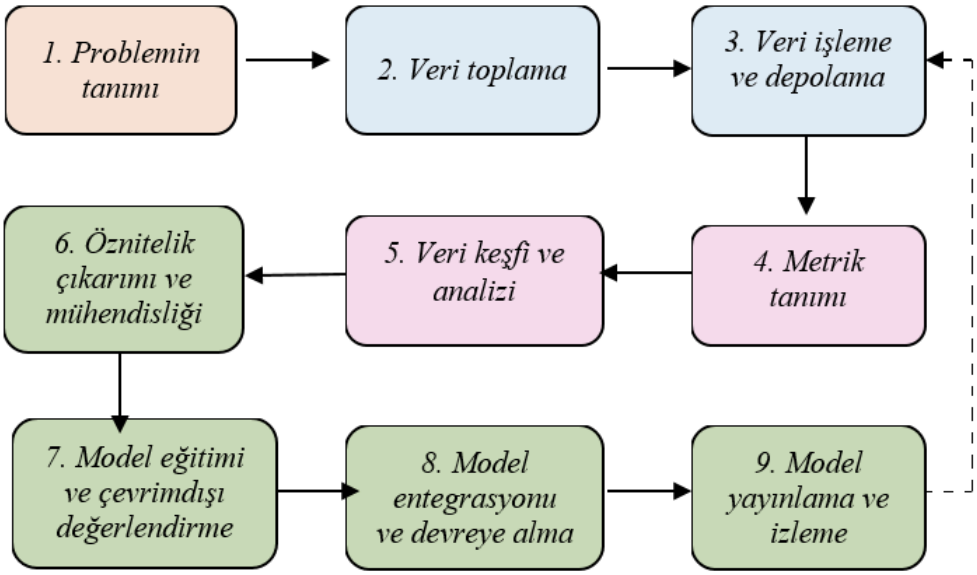



**Şekil 2.1:** CRISP-DM adımları (Chapman ve diğ., 2000; Çakır, 2008).

Ancak CRISP-DM önerisinin günümüzdeki makine öğrenmesi çalışmalarının ihtiyaçlarını karşılamakta yetersiz kalması nedeniyle daha güncel yaklaşımların MLOps (makine öğrenmesi operasyonları - *machine learning operations*) başlığı altında model yaşam döngüsü *(model life cycle)* şeklinde incelenmekte olduğu görülmektedir. Gunning ve diğerlerinin (2019) makine öğrenmesi çalışmalarının uçtan uca değerlendirilmesine yönelik tanımladıkları model yaşam döngüsü adımları;

1- Problemin tanımı
2- Veri toplama
3- Veri işleme ve depolama
4- Metrik tanımı
5- Veri keşfi ve analizi
6- Öznitelik çıkarımı ve mühendisliği
7- Model eğitimi ve çevrimdışı değerlendirme
8- Model entegrasyonu ve devreye alma
9- Model yayınlama ve izleme

şeklindedir (Şekil 2.2). Yaşam döngüsü, bir sorun tanımlandığında başlar ve sonra kendi içinde geri beslemeli olarak devam eder. Bu durum, model geliştirmenin yinelemeli doğası olarak adlandırılır. Bu döngünün birkaç iterasyon çalışması durumunda, analiz edilmesi veya toplanması gereken veriler zaten mevcut olabilir ve bazı adımları atlamak mümkün olabilir. Sürekli değişen veri ve gereksinimlere ayak uydurmak için döngü kendini tekrar eder. Model performans yönetimi, bir makine öğrenimi modelinin eğitimi, devreye alınması ve izlenmesi arasındaki döngüyü kapsar. Bu kapsam sayesinde, değişiklikleri ve sapmaları tespit etmeye yardımcı olabilir ve analitik ekiplerin hızla güncelleme yapmasını sağlayabilir (Gunning ve diğ., 2019).




**Şekil 2.2:** Model yaşam döngüsü adımları (Gunning ve diğ., 2019).

Veri kümelerinin analizinde farklı makine öğrenmesi yaklaşımları kullanılmaktadır. Makine öğrenmesi yaklaşımları temel olarak danışmanlı (gözetmenli veya denetimli) ve danışmansız (gözetmensiz veya denetimsiz) olmak üzere ikiye ayrılmaktadır (Hastie ve diğ., 2009). Ancak literatürde yer alan bazı kaynaklarda danışmanlı ve danışmansız öğrenmeye ek olarak yarı danışmanlı öğrenme, topluluk öğrenme ve pekiştirmeli öğrenme yöntemlerine de yer verilmektedir (Alpaydin, 2014; Mohri ve diğ., 2018; Burkov, 2019).

Danışmanlı öğrenme yaklaşımı, literatürde sıklıkla kullanılan makine öğrenmesi yöntemlerinden birisi olup, bu yaklaşımda etiketlenmiş olan örneklerden oluşan bir veri kümesi kullanılarak öğrenme gerçekleşir (Akhtar, 2017). Sınıflandırma ve regresyon danışmanlı öğrenme başlığı altındaki makine öğrenmesi fonksiyonlarıdır. Girdi vektörü olarak sonlu sayıdaki ayrık kategoriden birisine atama yapılması durumunda sınıflandırma, istenen çıktı bir veya daha fazla sayıda sürekli değerden oluşması durumunda ise regresyon yöntemi adı verilmektedir (Bishop, 2006).

Sınıflandırma analizinde kategorik bir hedef değişken vardır ve bu hedef değişkenin her bir farklı değeri sınıf etiketi olarak adlandırılır. Sınıflandırma analizi, gerçekleştirilmiş ve sınıf etiketi atanan verilerden yola çıkarak sınıf etiketi olmayan her örneğe doğru sınıf etiketi atamayı amaçlayan bir tahmin modeli geliştirme çalışmasıdır. Sınıflandırmaya örnek olarak, bir bankanın müşterilerinin kredi başvurularını değerlendirme aşamasında, başvuran müşterinin kayıtları ile daha önceki müşterilerinin kayıtlarını inceleyerek kredi başvurusuna olumlu ya da

olumsuz bir sonuç iliştirmesi işlemi verilebilir. Bu örnekte, olumlu yanıt ve olumsuz yanıt, sınıf etiketleridir. Sınıflandırma işlemi, başvuran müşterinin verilerinin bankanın önceki müşterileri üzerinden oluşturulan modele göre değerlendirilerek uygun etiketin belirlenmesidir (Alpaydin, 2014).

Danışmanlı bir öğrenme algoritmasının amacı, veri kümesini girdi olarak bir öznitelik vektörü alan ve bu öznitelik vektörü için etiketin bulunmasına imkan sağlayan bilgileri çıkaran bir model üretmek için kullanmaktır (Burkov, 2019).

Sınıflandırma modeli, eğitim verilerinin birden çok analizinden türetilmiştir. Sınıflandırma çalışmasının modeli, eğer – o zaman (IF – Then) kuralları, matematiksel fonksiyonlar, karar ağaçları veya sinir ağları gibi birçok farklı yapıda çıktı verebilir. Veritabanları, karar verme süreçlerinde kullanılabilecek örtük bilgileri içermesi açısından oldukça zengindir. Sınıflandırma, veri sınıflarını tanımlayan ve veriye dayalı olarak gelecekteki eğilimlere yönelik tahmin modelleri çıktısı veren bir analiz biçimidir. Sınıflandırma analizi, büyük verilerin daha iyi anlaşılmasını sağlayabilir. Sınıflandırma analizi, büyük veri kümeleri üzerinde makine öğrenimi çalışmaları kapsamında oldukça yaygın şekilde kullanılmaktadır (Han, Kamber, ve Pei, 2012).

Sınıflandırma modelini değerlendirmek için hata matrisi, sınıflandırma hata oranı, kappa istatistiği, ROC eğrisi, F1 skor, ortalama mutlak hata, mutlak bağıl hata, bağıl hataların karekökü gibi yaklaşımlar kullanılabilir (Witten ve diğ., 2011; Akpınar, 2014).

Sınıflandırma çalışmalarında, iki etiketli sınıf barındıran veri kümeleri için varlığı tahmin edilmeye çalışılan etiketin pozitif diğerinin ise negatif olarak adlandırılır. Sınıflandırma tahmin modeli tarafından pozitif sınıf için doğru tahmin edilen örneklere doğru pozitif (TP), yanlış tahmin edilenler için ise yanlış pozitif (FN) ifadesi kullanılır. Benzer şekilde, negatif etiketliler için de doğru negatif (TN) ve yanlış negatif (FN) olarak adlandırılır. Bu tanımlamaya dayalı olarak sınıflandırma doğruluğu Denklem (2.1) ile ifade edilmektedir.

$$\text{Sınıflandırma doğruluğu} = \frac{\text{TN} + \text{TP}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \tag{2.1}$$

Sınıflandırma doğruluğu, sınıflandırma çalışmalarında yaygın olarak kullanılan bir değerlendirme ölçüsüdür ancak her zaman tek başına kullanımında sonuçların başarısını açıklamada yeterli olamayabilmektedir. Sınıflandırma doğruluğu ile birlikte kullanılan yaygın kullanılan değerlendirme ölçülerinden bir diğeri olan F1 skor Matematiksel olarak Denklem (2.2)'deki şekilde ifade edilmektedir (Witten ve diğ., 2011).

$$\text{F1 skor} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}} \tag{2.2}$$

Literatürde yaygın olarak kullanılan sınıflandırma algoritmalarını;

- Yapay Sinir ağları *(Neural Networks)* (McCulloch ve Pitts, 1943)
- Stokastik gradyan iniş *(Stochastic Gradient descent - SGD Classifier)* (Bottou, 1991)
- Çok katmanlı algılayıcılar *(Multi-Layer Perceptron- MLP Classifier)* (Baxter ve diğ., 2011)
- Lojistik Regresyon (Hart ve diğ., 2000)
- Karar destek makinesi sınıflandırıcısı *(Support Vector Machine Classifier)* (Guyon ve diğ., 2002)
- Pasif agresif sınıflandırıcı *(Passive Aggressive Classifier)* (Lu ve diğ., 2016)
- Rastgele orman *(Random Forest Classifier)* (Breiman, 2001)
- Gradient Boosting Classifier (Friedman, 2001)
- Ada Boost Classifier (Freund ve Schapire, 1996)
- Ekstra Ağaçlar (*Extra Trees Classifier*) (Geurts ve diğ., 2006)
- K – en yakın komşuluk *(K-Neighbors Classifier)* (Cover ve Hart, 1967)
- Naïve Bayes (Bayes ve Price, 1763)
- Karar Ağacı *(Decision Tree Classifier)* (Hart ve diğ., 2000)
- CatBoost (Prokhorenkova ve diğ., 2018)

şeklinde ifade edilebilir.

Danışmansız öğrenme, herhangi bir hedef değişkenin olmadığı veya bilinemediği durumlarda, sadece girdi değerlerinden hareket edilerek eldeki girdilerin gruplanması çalışmalarıdır (Balaban ve Kartal, 2018). Danışmansız öğrenmede amaç, çıktı değeri olmaksızın girdi değerleri arasındaki ilişki ve örüntüleri tanımlayabilmektir (Hastie ve diğ., 2009).

Genel olarak danışmansız öğrenmenin uzun ve etkili bir tarihi vardır. Bu anlamda bazı ilk dönem çalışmaları incelendiğinde, Barlow'un (1989) nöral kodları karakterize etmenin yollarını araştırdığı, MacKay'in (1956) ise sibernetik – teorik bir yaklaşımı benimsediği görülmüştür. Öznitelik çıkarma *(feature extraction)* yöntemleri ise genellikle daha az kapsamlı bir şekilde araştırılmıştır (Dayan, 1999).

Danışmansız öğrenmede, veri kümesi etiketlenmemiş örneklerden oluşan bir koleksiyondur. Danışmansız bir öğrenme algoritmasının amacı, özellik vektörünü girdi olarak alan ve onu başka bir vektöre veya pratik bir sorunu çözmek için kullanılabilecek bir değere dönüştüren bir model oluşturmaktır. Örneğin, x girdi vektörü olmak üzere, kümelemede model veri kümesindeki her özellik vektörü için kümenin kimliğini döndürür. Boyut indirgemede, modelin çıktısı x girdisinden daha az özelliğe sahip bir özellik vektörüdür; aykırı değer tespitinde çıktı, x'in veri kümesindeki sıradan bir örnekten ne kadar farklı olduğunu gösteren gerçek bir sayıdır (Burkov, 2019).

Son zamanlarda araştırmacıların sınıflandırma amacı taşıyan çalışmalarında, etiketlenmiş ve etiketlenmemiş örneklerin bir arada olduğu durumlarda yarı danışmanlı öğrenme olarak adlandırılan alanda çalıştığı görülmektedir. Etiketlenmiş veri olmadan sınıflandırma yapılamamaktadır ancak az miktardaki etiketlenmiş veriyi büyük miktarda etiketlenmemiş veri havuzunda çoğaltmak etkili bir yaklaşım olabilir (Frank ve Hall, 2011).

Yarı danışmanlı öğrenmede, veri kümesi hem etiketli hem de etiketsiz örnekler içerir. Genellikle, etiketlenmemiş örneklerin sayısı, etiketli örneklerin sayısından çok daha fazla olduğunda kullanılması uygundur. Yarı danışmanlı öğrenme algoritmasının amacı, danışmanlı öğrenme algoritmasının amacı ile aynıdır. Buradaki amaç, birçok etiketlenmemiş örneğin kullanılmasının, öğrenme algoritmasının daha iyi bir model bulmasına yardımcı olabilmesidir. Öğrenmenin daha fazla etiketlenmemiş örnek eklemekten fayda sağlayabileceği sezgisel olarak düşünülebilir ancak bu durum çözülmek istenen sorunun belirsizliğini arttırmaktadır. Bununla birlikte, etiketlenmemiş örnekler eklendiğinde, problem hakkında daha fazla bilgi eklenmiş olur ve bu sayede daha büyük bir örnek oluşması ile etiketlenen örneklerin geldiği olasılık dağılımını daha iyi yansıtır (Burkov, 2019).

Ödül ve cezaya dayanan ve optimal karar verme süreçlerine temel oluşturan pekiştirmeli öğrenme yaklaşımı Sugiyama (2015) tarafından önerilmiştir. Pekiştirmeli öğrenme, iki temel makine öğrenmesi yaklaşımından farklılaşmaktadır. Bu yaklaşım, ne danışmanlı öğrenmedeki gibi

etiketlenmiş veriye dayanır ne de gizlenmiş bir yapıyı ortaya çıkarmaya çalışır. Burada asıl amaç, ödül sinyalinin *(reward signal)* maksimize edilmesidir (Sutton ve Barto, 2018). Alpaydin (2014) tarafından belirtilen "eleştirmenle öğrenme" ifadesindeki eleştirmen, danışmandan farklı olarak ne yapılması gerektiğini değil geçmiş eylemin iyi olup olmadığını söylemektedir. Pekiştirmeli öğrenme, belirli bir etikete ihtiyaç duymadan, geçici etiketler üzerinden deneme yanılma yolu ile öğrenme sürecidir (Balaban ve Kartal, 2018).

Topluluk öğrenmesi, tek bir süper modeli üretmek yerine, çok sayıda düşük doğruluklu modeli eğitmeye ve ardından bu zayıf modellerin alacağı tahminleri yüksek doğruluklu bir meta modeli elde etmek için birleştirmeye odaklanan bir öğrenme yaklaşımıdır. Temel makine öğrenmesi algoritmaların sınırlamaları vardır. Basit olmaları nedeniyle bazen problemler için yeterince doğru bir model üretemezler. Daha iyi sonuçlar elde edebilmek için derin sinir ağlarının kullanımı denenebilir. Bu durumda ise, derin sinir ağlarının oldukça büyük miktarda etiketlenmiş veri gerektirmesi ve elde bu boyutta veri bulunmaması durumu söz konusu olabilir. Derin öğrenme ile iyi sonuçların alınamayacağı bu durumlarda, basit öğrenme algoritmalarının performansını artırmak için kullanılabilecek başka bir yaklaşım topluluk öğrenmesi yaklaşımıdır. Düşük doğruluğa sahip modeller genellikle zayıf öğrenenler tarafından, yani karmaşık modelleri öğrenemeyen öğrenme algoritmaları tarafından öğrenilir ve bu nedenle genellikle eğitim ve tahmin için çalışma süreleri düşüktür. En sık kullanılan zayıf öğrenici, yalnızca birkaç yinelemeden sonra eğitim setini bölmeyi bırakan bir karar ağacı öğrenme algoritmasıdır. Elde edilen ağaçlar sığdır ve özellikle doğru değildir, ancak topluluk öğrenmenin arkasındaki fikir, eğer ağaçlar aynı değilse ve her ağaç rastgele tahminden en azından biraz daha iyiyse, o zaman çok sayıda bu tür ağacı birleştirerek yüksek doğruluk elde edebileceği şeklindedir. Tahmin elde etmek için, her bir zayıf modelin tahminleri bir tür ağırlıklı oylama kullanılarak birleştirilir. Kullanılacak oy ağırlıklandırma biçimi algoritmaya bağlıdır, ancak algoritmadan bağımsız olarak fikir aynıdır. Eğer zayıf modeller topluluğu, mesajın spam olduğunu tahmin ederse, spam etiketi atanmasına dayalı bir çalışma prensibi söz konusudur (Burkov, 2019). Rastgele orman, en yaygın kullanılan topluluk öğrenme algoritmalarından bir tanesidir.

Pekiştirmeli öğrenme, makinenin bir ortamda yaşadığı ve o ortamın durumunu bir özellikler vektörü olarak algılayabildiği bir makine öğrenimi alt alanıdır. Makine her durumda eylemleri (*actions*) yürütebilir. Farklı eylemler, farklı ödüller getirir ve makineyi ortamın başka bir durumuna da taşıyabilir. Pekiştirmeli öğrenme, denetimli öğrenmedeki modele benzer şekilde,

bir durumun özellik vektörünü girdi olarak alan ve bu durumda yürütülecek en uygun eylemi çıkaran bir fonksiyondur. Bu fonksiyon, beklenen ortalama ödülü maksimize ediyorsa optimaldir. Pekiştirmeli öğrenme, oyun oynama, robotik, kaynak yönetimi veya lojistik gibi karar vermenin sıralı ve hedefin uzun vadeli olduğu belirli bir tür sorunu çözer (Burkov, 2019).

Ham verilerin elde edilmesinin ardından makine öğrenmesi çalışmaları için ön işleme aşamalarından geçirilerek analize hazır hale getirilmesi gerekir. Analizin sağlıklı bir şekilde sonuç verebilmesi için veri ön işleme çalışmalarının yapılması önemlidir. Eksik, gürültülü ve tutarsız veriler, büyük gerçek dünya veritabanlarının ve veri ambarlarının olağan özellikleridir. Eksik veriler birkaç nedenden dolayı ortaya çıkabilir. Bir yanlış anlama veya ekipman arızası nedeniyle ilgili veriler kaydedilemeyebilir. Kaydedilen diğer verilerle tutarsız olan veriler silinmiş olabilir. Ayrıca, geçmişin kaydedilmesi veya verilerde yapılan değişiklikler gözden kaçmış olabilir (Han ve diğ., 2012). Veri ön işleme çalışmaları eksik değer atama, aykırı değer analizi, kesikleştirme, boyut indirgeme ve normalizasyon gibi çalışmaları kapsar.

Çeşitli nedenlerle kayıp değerlerin bulunduğu veri kümeleri üzerinde, makine öğrenmesi algoritmaları uygulanmadan önce eksik değer ataması yapılır. Eksik değer atamasında, ortalama, mod ve medyan gibi istatistiki yaklaşımlar kullanılarak atamalar yapılabilir. Ayrıca eksik değer barındıran sürekli nitelikler için regresyondan faydalanılarak değer ataması yapılabilmektedir. Kategorik nitelikler için eksik değer atamasında sınıflandırma algoritmaları kullanılabilir (Sezer ve Başeğmez, 2022).

Aykırı değerler, bir veri kümesinde normalin dışında kalan, sahip oldukları özelliklerle diğer nesnelerden aşikar şekilde aykırılık gösteren nesnelerdir. Aykırı değer tespitinde standart sapma, kartiller arası uzaklıklar ve kümeleme algoritmaları kullanılabilir (Akpınar, 2014).

Sürekli değişkenlerin barındırdığı değerleri kategorik değerlere dönüştürme işlemine kesikleştirme adı verilir (Özkan ve Erol, 2015). Sadece kategorik verilerle çalışabilen algoritmaların ve fonksiyonların kullanılabilmesi için kesikleştirme işlemi yapılmasına ihtiyaç duyulabilir.

Veri kümesinin barındırdığı karmaşayı azaltmak için özniteliklerin bir kısmının seçilmesine öznitelik alt kümesi seçimi (*feature subset selection),* eldeki niteliklerin birleştirilerek yeni niteliklerin oluşturulmasına öznitelik çıkarımı (*feature extraction)* adı verilir. Bu çalışmalar genel olarak boyut indirgeme başlığı altında incelenir (Alpaydin, 2014).

Normalizasyon, sürekli değişkenlerin barındırdığı değerlerin [0,1] ya da [-1,1] aralığına sıkıştırılması işlemidir. Bu indirgeme işlemleri için Z-skor ve min-maks yaklaşımlarından faydalanılabilir (Akpınar, 2014).

## 2.2 ÖZNİTELİK SEÇİMİ

Sınıflandırma çalışmalarında, elde bulunan nitelikler arasından sınıf belirlendikten sonra kalan diğer niteliklere sınıfın öznitelikleri adı verilir (Han ve diğ., 2012; Alpaydin, 2014).

Sınıflandırma problemlerinde, bilgi içerebileceğine inanılan tüm gözlem verileri çalışmaya girdi olarak alınır. Kusursuz bir dünyada yaşasaydık, bu girdilerin arasından önemli olanların seçilmesine gerek olmazdı ve sınıflandırma modeli kendisi gereksiz öznitelikleri atabilirdi. Öznitelik seçimi ile, gereksiz girdilerin toplanması maliyetinden tasarruf edilmesi ve veri içerisindeki gürültünün azaltılması sağlanabilir (Alpaydin, 2014).

Öznitelik seçimi, makine öğrenimi çalışmalarının veri ön işleme sürecinde en sık kullanılan önemli tekniklerden birisidir ve zamanla makine öğrenimi sürecinin vazgeçilmez bir bileşeni haline gelmiştir (Kalousis ve diğ., 2007). Makine öğrenimi ve istatistikte değişken seçimi, özellik seçimi veya değişken alt küme seçimi olarak da bilinir. İlgili öznitelikleri tespit etme ve alakasız, gereksiz veya gürültülü verileri kaldırma işlemidir. Bu süreç, veri madenciliği algoritmalarını hızlandırır, tahmin doğruluğunu geliştirir ve anlaşılırlığı artırır. Alakasız öznitelikler, yararlı bilgi sağlamayan özniteliklerdir ve gereksiz öznitelikler, o anda seçili olan özniteliklerden daha fazla bilgi sağlamaz (Molina ve diğ., 2002). Öznitelik alt kümesi seçiminde amaç eldeki özniteliklerin en iyi alt kümesini bulmaktır. En iyi alt küme, sınıflandırma başarısını en yüksek yapan en az sayıda boyuttan oluşur (Alpaydin, 2014).

Sınıflandırma için öznitelik seçimi, sonuçlarına göre ve içinde sınıflandırıcı barındırıp barındırmamasına göre alt kırılımlara ayrılarak incelenmektedir. Sonuçlarına göre öznitelik seçimi yaklaşımları, bir öznitelik alt kümesi sonucunu döndürenler ve öznitelikleri sıralayanlar olarak iki gruba ayrılır. Sınıflandırıcı barındırmasına göre yapılan ayrımda ise, öznitelik seçimi yapılan aşamada bir sınıflandırıcıdan faydalanılıyorsa sarmalayıcı (*wrapper*) yaklaşım, sınıflandırıcı kullanılmadan sonuç elde ediliyorsa da filtreleyici *(filter*) yaklaşım olarak iki gruba ayrılır. Ayrıca öznitelik seçimi yaklaşımlarının kullandıkları ölçüler ve arama yaklaşımlarına göre de ayrımlar mevcuttur (Liu ve Motoda, 1998b).

Bu tez çalışmasında, öznitelik seçim yaklaşımları, Liu ve Motoda'nın (1998b) çalışmasına paralel olarak; sonuçlarına, arama yaklaşımlarına, kullanılan ölçülere ve sınıflandırıcı barındırmasına göre dört açıdan ele alınmaktadır (Şekil-2.3).




**Şekil 2.3:** Öznitelik seçimine bakış açıları

### 2.2.1 Sonuçlarına Göre Öznitelik Seçimi

Sonuçlarına göre öznitelik seçimi yaklaşımları alt küme sonucu döndüren ve özniteliklerin sıralanmış bir listesini döndüren yaklaşımlar olarak ikiye ayrılır.

#### *2.2.1.1 Alt Küme Sonucu Döndüren Yaklaşımlar*

Öznitelik alt kümesi seçim yaklaşımları, çalışma içerisinde belirlenen arama yaklaşımına dayalı olarak değerlendirilen tüm aday öznitelik alt kümeleri için birer skor üretirler ve bu skorlara dayalı olarak belirlenen daraltılmış bir öznitelik alt kümesi döndürürler. Öznitelik alt kümesi seçim yaklaşımları ile elde edilen öznitelik alt kümesi ise direk olarak tahmin çalışmalarının yapılmasına olanak sağlamaları sayesinde başka herhangi bir yaklaşıma ihtiyaç duymazlar. Ancak öznitelik alt kümeleri, seçim aşamasında kullanılan merit değerine göre skorladığı için, seçilmiş olan öznitelik alt kümesi içerisindeki öznitelikler için bireysel herhangi bir çıktı üretemezler. Ayrıca n adet özniteliğe sahip bir veri kümesi için, öznitelik alt kümesi uzayının boyutu $2^n$ büyüklüğünde olduğu için, eldeki değerlendirilebilecek olası aday sayısı her zaman öznitelik sayısından büyüktür. n değeri büyüdükçe $2^n$ değeri de üstel bir artış göstereceğinden

dolayı aradaki fark da büyüyerek artacaktır. Bu nedenle öznitelik alt kümesi yaklaşımları kullanılırken sezgisel arama (*heuristic search*) veya rastgele arama (*random search*) gibi yaklaşımlardan faydalanılarak tüm öznitelik alt kümesi uzayını skorlamak yerine nispeten daha küçük bir öznitelik alt uzayını skorlayarak daha az maliyetle çözüm üretilebilmektedir. Öznitelik alt uzayında sezgisel veya rastgele arama yapan yaklaşımlar, en iyi öznitelik alt kümesini bulmayı garanti etmezler (Sezer ve Çakır, 2022).

#### *2.2.1.2 Öznitelik Sıralaması Döndüren Yaklaşımlar*

Öznitelik sıralama yaklaşımları (*feature ranking approaches*), sonuç olarak eldeki tüm öznitelikleri, bireysel (*individual)* olarak değerlendirme merit değeri cinsinden skorlar ve bu skor değerine göre sıralamaya olanak sağlar. Açıklanabilir yapay zeka (*Explainable AI*) yaklaşımlarında da öznitelik önem (*feature importance*) değeri üzerinden sıralama çıktıları ortaya çıkarılarak, elde edilen tahmin (*prediction*) modellerinin anlamlandırılması üzerinde çalışılır. Benzer şekilde, öznitelik sıralama algoritmaları da araştırmacıya eldeki özniteliklerin her biri için bireysel olarak özniteliğin skorunu ve diğer özniteliklere göre durumunu gösterdiği için aynı zamanda açıklanabilir bir çıktı ürettiği de söylenebilir. Ancak, öznitelik sıralama algoritmaları sonuç olarak net bir öznitelik alt kümesi vermediği için, sıralanmış olan özniteliklerden bir kısmını seçmek için bir eşik değer (*threshold*) belirleme yaklaşımına ya da bir başka öznitelik alt kümesi seçim yaklaşımına ihtiyaç duyarlar. Öznitelik sıralayan yaklaşımlar, sonuç olarak sıralanmış bir öznitelik listesi döndürürler (Liu ve Motoda, 1998b).

### 2.2.2 Kullanılan Ölçülere Göre Öznitelik Seçim Yöntemleri

Kullanılan ölçülere göre öznitelik seçim yöntemleri, enformasyon (*information*), korelasyon (*correlation*), tutarlılık (*consistency*) ve sınıflandırma hata oranı (*prediction error rate*) olmak üzere beş farklı başlık altında incelenmektedir (Liu ve Motoda, 1998b).

Öklid, Minkowski, Manhattan ve Hamming başlıca kullanılan mesafe ölçüleridir. Öznitelikler arasında hiç uzaklık bulunmadığında, bu öznitelikler birbirlerinden ayırt edilemezler. Üzerinde çalışılan veri kümesinde alakasız özniteliklerin sayısının fazla olması, mesafe ölçüsü yaklaşımını kullanan öznitelik seçimi yaklaşımları için yanıltıcı olabilir (Liu ve Motoda, 1998b; Özkan ve Erol, 2015).

Enformasyon ölçüsü kullanan öznitelik seçim yaklaşımları genel olarak, üzerinde öznitelik seçimi yapılan veri kümesinde bulunan öznitelikler ile bilgi kazanımı (*information gain*) hesaplaması yapılmasına dayanır ve yüksek bilgi kazanımı elde edilen özniteliklerin seçilmesi prensibi ile çalışır (Liu ve Motoda, 1998b).

Öznitelik seçiminde korelasyon ölçülerinin kullanımı, sınıf ile yüksek korelasyona sahip olan özniteliklerin seçilmesi ve sınıf ile düşük korelasyona sahip özniteliklerin elenmesi prensibine dayanır (Dash ve Liu, 2003).

Öznitelik seçiminde doğrusal korelasyon ve enformasyon teorisinden faydalanan yaklaşımlar olmak üzere iki farklı kullanım literatürde mevcuttur. Doğrusal korelasyon öznitelik seçimi için oldukça iyi bir yaklaşım olmasına karşın, kategorik ve sürekli öznitelikleri barındıran karma veri kümelerinde çalışabilmek için simetrik belirsizlik ve enformasyon teorisinden faydalanan yaklaşımların kullanımı tercih edilmektedir (Yu ve Liu, 2003).

Kategorik ve sürekli öznitelikleri birlikte barındıran karma veri kümelerinde öznitelik seçimi üzerine çalışılırken, bilgi kazanımı ve simetrik belirsizlik gibi hesaplamaların yapılabilmesi için sürekli özniteliklerin kesikleştirilmesi ve kaba kümelerden (*rough sets*) faydalanılmaktadır (Jiang ve Wang, 2016).

Tutarlılık değerlendirme ölçülerinin diğerlerinden farklı yönleri bulunmaktadır. Tutarlılık ölçüsünü kullanan öznitelik seçimi yaklaşımlarının, diğer ölçüleri kullanan yaklaşımların öğrenme verisine (*train set*) aşırı bağımlılığı ve küçük bir öznitelik alt kümesi oluşturma yanlılığına karşı daha dirençli olduğu söylenebilir. Tutarlılık ölçüleri, tutarsızlık değerinin minimize edilmesi prensibine dayalı olarak çalışır (Liu ve Motoda, 1998b). Tutarsızlık hesaplaması üç aşamada ifade edilebilir (Dash ve Liu, 2003);

1) s1 ve s2 sınıf etiketlerine ait iki farklı değer olmak üzere, (0 1 0 s1) ve (0 1 0 s2) gibi açıklayıcı özniteliklerin aynı değerleri aldığı ancak sınıfın farklı değer aldığı örnekler tutarsız olarak değerlendirilir.

2) (0 1 0 s1), (0 1 0 s2), (0 1 0 s3) şeklinde açıklayıcı özniteliklerin aynı değeri aldığı ve sınıf etiketinin farklı olduğunda, sınıfın s1, s2, s3 değerlerinin veri kümesi içerisinde tekrarlanma sayıları k1, k2, k3 şeklinde gösterilebilir. Örnek sayısı n olmak üzere, n=

k1+k2+k3 olarak ifade edilebilir. k3 değerini k1 ve k2 değerinden büyük olduğu varsayılırsa, n - k3 değerine tutarsızlık adedi adı verilir.

3) Veri kümesinde bulunan her bir açıklayıcı öznitelik kombinasyonu için tutarsızlık adedinin toplamının, veri kümesinde bulunan öznitelik kombinasyonu adedine bölünmesi ile tutarsızlık oranı hesaplanır. Bu oranın 1'den çıkarılması ile tutarlılık oranı hesaplanabilir (Liu ve Motoda, 1998b; Sezer, 2018).

Sınıflandırıcı hata oranı *(predictive error rate)* metriğini kullanabilmek için öznitelik seçimi aşamasında bir sınıflandırma algoritmasının kullanılması gerekmektedir. İteratif şekilde tüm aday öznitelik alt kümelerine sınıflandırma algoritması uygulanır ve en düşük sınıflandırıcı hata oranına sahip adayın seçilmesi sağlanır. Öznitelikler seçildikten sonra sınıflandırıcı olarak kullanılarak sınıflar tahmin edilir. Sınıflandırma doğruluğunun çok yüksek seviyeye ulaşması ile birlikte diğer yöntemlere kıyasla hesaplama yükü de doğru orantılı bir şekilde artarak çok yüksek seviyelere ulaşabilmektedir (Liu ve Motoda, 1998b; Dash ve Liu, 2003).

### 2.2.3 Öznitelik Seçiminde Arama Yöntemleri

Öznitelik alt küme uzayının boyutu, n adet öznitelikten oluşan bir veri kümesi için $2^n$ büyüklüğünde olmaktadır ve n değeri büyüdükçe alt küme uzayı oldukça hızlı büyümektedir. Günümüz çalışmalarında, veri kümeleri oldukça büyük hacimlere ulaşabildiği için eldeki tüm alternatifleri değerlendirmek her zaman kolay olmayabilmektedir. Bu nedenle, bazı sınırlı arama yaklaşımları geliştirilmiştir. Bu yöntemlerden en yaygın olarak kullanılanları bu kısımda detaylı olarak açıklanmıştır (Liu ve Motoda, 1998b; Sezer ve Çakır, 2022).

Tam arama (*exhaustive search*) yöntemi, öznitelik alt kümesi uzayının tamamını değerlendirerek çalışır. Bu yaklaşım, öznitelik sayısının çok olduğu durumlarda oldukça fazla işlem maliyeti gerektirebilir (Liu ve Motoda, 1998b; Sezer ve Çakır, 2022).

Sezgisel arama (*heuristic search*), evrimsel algoritmalar gibi bazı optimizasyon algoritmalarının değerlendirme sürecine dahil edilmesi prensibine dayalı arama yaklaşımlarıdır. Çok amaçlı optimizasyon algoritmalarının da kullanımı son yıllarda oldukça artmıştır. Kullanılan başlıca sezgisel algoritmalar;

- Genetik algoritma (Holland, 1992)

- Baskın olmayan sıralı genetik algoritma *(Non-dominated sorting genetic algorithm)* (NSGA) (Srinivas ve Deb, 1994)
- Açgözlü algoritmalar *(Greedy Stepwise)* (Jacobs ve Brusco, 1995)
- Parçacık sürü optimizasyon algoritması *(Particle swarm optimization)* (Kennedy ve Eberhart, 1995)
- İkili guguk kuşu arama algoritması *(Binary Cuckoo search)* (Rodrigues ve diğ., 2013)
- Stokastik fraktal arama algoritması *(Stochastic fractal search)* (Salimi, 2015)
- En iyi öncelikli arama algoritması *(Best First search)* (Frank ve diğ., 2016)
- Şempanze optimizasyon algoritması *(Chimp optimization algorithm)* (Khisne ve Mosavi, 2020)
- İkili çok amaçlı şempanze optimizasyon algoritması (*Binary multi-objective chimp optimization algorithm*) (Piri ve diğ., 2022c)

şeklinde ifade edilebilir.

Rastgele Arama (*Random search*), çalışmaya dahil olan öznitelik uzayının içerisinden rastgele sayıda tekrarsız seçilmiş adayın değerlendirilmesi prensibine dayalıdır.

### 2.2.4 Sarmalayıcı ve Filtreleyici Yaklaşımlar

Öznitelik seçimi çalışmaları, içerisinde bir sınıflandırıcı barındırıp barındırmamasına göre sarmalayıcı veya filtreleyici olarak iki başlık altında incelenmektedir.

Sınıflandırma algoritmalarında kullanılan öznitelik alt kümesi seçme yöntemleri filtreleyici ve sarmalayıcı olmak üzere, içerisinde sınıflandırıcı barındırmasına göre iki başlık altında incelenmektedir. Sarmalayıcı yöntemler, içerisinde bir sınıflandırıcı bulundurur, değerlendirdiği her bir öznitelik alt kümesi için bir sınıflandırma çalışması gerçekleştirir ve değerlendirme ölçüsü olarak sınıflandırma hata oranını kullanır. Filtreleyici yöntemler ise, içerisinde bir sınıflandırma algoritması bulundurmadığından dolayı, alternatif yollardan ölçüm yaparlar ve sınıflandırma doğruluğundan farklı bir değerlendirme ölçüsü kullanmaktadırlar (Blum ve Langley, 1997; Sezer ve Çakır, 2022).

#### *2.2.4.1 Sarmalayıcı Yaklaşımlar*

Öznitelik seçme algoritmalarında sınıflandırıcının doğruluğunu bir performans ölçüsü olarak kullanmak basit bir yaklaşımdır. Mümkün olan en yüksek sınıflandırıcı doğruluğunu elde etmek için öncelikle bir sınıflandırıcı oluşturulmalı ve daha sonra sınıflandırıcıya en uygun öznitelikler seçilmelidir (Liu ve Motoda, 1998b). Bu yaklaşım, Kohavi ve John tarafından 1997 yılında ortaya atılmıştır ve sarmalayıcı (*wrapper*) yaklaşım olarak adlandırılmıştır (Kohavi ve John, 1997; Liu ve Motoda, 1998b).

Sarmalayıcı yaklaşımda, öznitelik alt küme uzayı üzerinde bulunan tüm aday alt kümeler için sınıflandırıcı kullanılarak, sınıflandırma hata oranına dayalı skorlama yapılır ve taranan adaylar arasından en yüksek performans (sınıflandırma başarısı) gösteren alt kümenin seçilmesi sağlanır (Kohavi ve John, 1997; Sezer, 2018).

#### *2.2.4.2 Filtreleyici Yaklaşımlar*

Sarmalayıcı yaklaşım dışında kalan, sınıflandırıcı barındırmayan yaklaşımlara filtreleyici yaklaşımlar (*filter approaches*) adı verilir. Filtreleyici yaklaşımlar, enformasyon (*information*), mesafe (d*istance*), korelasyon (*correlation*) ve tutarlılık (*consistency*) değerlerini merit olarak kullanır. Dolayısıyla, filtreleyici yaklaşımlar tahmin doğruluğu (*predictive accuracy*) değerini maksimize etmeyi garanti etmezler ancak sarmalayıcı yaklaşıma göre nispeten daha az işlem yüküne ve süreye ihtiyaç duyarlar (Liu ve Motoda, 1998b).

### 2.2.5 Öznitelik Seçim Algoritmaları

Literatürde bilinen çok sayıda öznitelik seçim algoritması bulunmaktadır. Bu tez çalışmasında, Korelasyon Bazlı Öznitelik Seçimi (CFS) ve Tutarlılık Bazlı Öznitelik Seçimi Yöntemi (CONS) yaklaşımları kullanılmıştır.

#### *2.2.5.1 Korelasyon Bazlı Öznitelik Seçimi*

Korelasyon bazlı öznitelik alt kümesi seçimi motivasyonu, Hall (1999) tarafından şu şekilde ifade edilmiştir:

*"İyi bir öznitelik alt kümesinin barındırdığı öznitelikler, sınıf ile yüksek korelasyona sahip iken kendi aralarında korelasyon bulunmayanlardır."*

Korelasyon bazlı öznitelik seçimi (*correlation based feature selection* - CFS), korelasyona dayalı sezgisel bir değerlendirme işlevi kullanarak öznitelik alt kümelerini sıralayan temel bir filtre algoritmasıdır. Değerlendirme fonksiyonu, sınıfla büyük ölçüde ilişkili ancak birbiriyle ilişkili olmayan özniteliklere sahip alt kümeleri tercih eder. Alakasız özniteliklerin sınıfla düşük korelasyona sahip olması göz ardı edilmelidir. Kalan özniteliklerden biri veya birkaçıyla yoğun bir şekilde ilişkili olacak gereksiz öznitelikler kaldırılmalıdır (M.A. Hall, 2000). CFS yöntemi, aşağıdaki gibi tanımlanan bir D kümesi için, $1 \leq k \leq n$ olmak üzere, veri kümesindeki k özniteliklerinin bir S alt kümesinin kalitesini ölçmek için $\Phi_D(S)$ işlevi Denklem (2.3) ile gösterilmiştir:

$$\Phi_D(S) = \frac{k.\sigma_D^C}{\sqrt{k + k.(k-1).\sigma_D^f}} \tag{2.3}$$

Burada $\sigma_D^C$, S'deki her bir öznitelik ile sınıf özniteliği arasındaki korelasyonların ortalamasıdır ve $\sigma_D^f$, S'deki $\binom{k}{2}$ olası öznitelik çiftlerinin her biri arasındaki korelasyon değerlerinin ortalamasıdır.

CFS, kategorik özniteliklerle çalıştığı için sürekli özniteliklerin kesikleştirilmesi sağlanır. Kesikleştirme işlemi için genellikle Fayyad ve Irani (1993) tarafından önerilen yaklaşım kullanılır.

CFS içerisinde korelasyon hesaplaması için, öznitelikler kümesi A, sınıf değerleri kümesi C olmak üzere, Denklem (2.4) ve Denklem (2.5) ile hesaplanan entropi değerleri kullanılarak Denklem (2.6) ile simetrik belirsizlik hesaplanır (Hall ve Holmes, 2000).

$$\mathrm{H(C)} = -\sum_{\mathrm{c \in C}} \mathrm{P(c)}\ \log_2 \mathrm{P(c)} \tag{2.4}$$

$$\mathrm{H(C|A)} = -\sum_{\mathrm{a \in A}} \mathrm{P(a)} \sum_{\mathrm{c \in C}} \mathrm{P(c|a)}\ \log_2 \mathrm{P(c|a)} \tag{2.5}$$

$$\mathrm{SU(X,C)} = 2 \times \left[\frac{\mathrm{H(X)} + \mathrm{H(C)} - \mathrm{H(X,C)}}{\mathrm{H(X)} + \mathrm{H(C)}}\right] \tag{2.6}$$

#### *2.2.5.2 Tutarlılık Bazlı Öznitelik Seçimi*

Tutarlılık ölçüsü, veri kümesini çok baskın bir sınıfa sahip bölümlere ayıran öznitelikleri bulmak için tasarlanmıştır. Bu ölçüm ilk olarak, Almuallim ve Dietterich (1991) ile Liu ve Setiono (1996) tarafından tartışılmıştır. Ancak, Dash ve Liu (2003) daha kapsamlı bir görüş sunmaktadır. Bölüm 2.2.2'de tutarsızlık hesaplamasına ait aşamalar detaylı olarak ifade edilmiştir.

Tutarlılık bazlı öznitelik seçim yöntemi, özniteliklerin sınıf ile tutarlılıklarının hesaplanması prensibine dayanır. Tutarlılık bazlı öznitelik seçimi yaklaşımı, sınıf ile yüksek tutarlılıkta ve küçük öznitelik alt kümeleri oluşturmaya meyillidir (Liu ve Motoda, 1998b; Sezer, 2018).

$S$ bir değişken alt kümesi, $S$ için oluşturulabilecek değişken alt küme sayısı $j$, veri kümesinin içerdiği satır sayısı $N$, satır numarası $i$, $i$-inci satırdaki değişkenin aldığı değerin meydana gelme sayısı $D_i$, $i$-inci satırdaki değişkenin aldığı değerin en geniş sınıftaki kardinalitesi $M_i$ olmak üzere, $S$ kümesine ait tutarlılık merit değeri Denklem (2.7)'deki şekilde hesaplanır:

$$CONS_S = 1 - \frac{\sum_{i=0}^{j} |D_i| - |M_i|}{N} \tag{2.7}$$

Korelasyon bazlı öznitelik seçimi (CFS) ve tutarlılık bazlı öznitelik seçimi (CONS) merit değerlerinin hesaplanmasından önce sürekli öznitelikler kesikleştirilir (Hall ve Holmes, 2003; Abraham ve diğ., 2009; Pedrycz ve diğ., 2016)

Tutarlılık metriği, tüm öznitelikleri kategorik halde bulunan bir veri kümesi için hesaplanabilmektedir. Varsa kategorik olmayan özniteliklerin tutarlılık hesaplaması öncesinde kategorikleştirilmesi sağlanır. Tamamı kategorik halde bulunan bir veri kümesi üzerinde yapılan tutarlılık hesaplaması, tüm kategorik özniteliklerin kukla (*dummy*) özniteliklere dönüştürülmüş hali ile yapılan K – en yakın komşuluk (KNN) hesaplaması ile birebir örtüşmektedir. Bu çıkarım, bu tez çalışması kapsamında yapılan analizlerde farkedilmiş olup, literatürde rastlanılmamıştır.

## 2.3 ÖZNİTELİK SEÇİMİ VE OPTİMİZASYON

Bir optimizasyon problemi, bir dizi bağımsız değişken veya parametre ile başlar ve genellikle değişkenlerin kabul edilebilir değerlerini tanımlayan koşulları veya kısıtlamaları içerir. Bu tür

kısıtlamalara problemin kısıtları adı verilir. Bir optimizasyon probleminin diğer temel bileşeni, bir şekilde değişkenlere bağlı olan ve amaç fonksiyonu olarak adlandırılan tek bir iyilik ölçüsüdür. Bir optimizasyon probleminin çözümü, amaç fonksiyonunun optimal bir değer varsaydığı değişkenlerin izin verilen değerleri kümesidir (Gill ve diğ., 1981).

Öznitelik seçme probleminde en uygun durumu bulmak için arama veya optimizasyon yöntemleri gereklidir. Optimizasyon, mevcut olanlar arasından mümkün olan en iyi çözümü bulmaya çalışma süreci olarak düşünülebilir. Başka bir ifade ile, eldeki kısıtlı kaynakların mevcut sınırlar ve koşullar çerçevesinde optimum kullanımını sağlamaktır (Kaya ve Fiğlali, 2016). Bu nedenle, optimizasyonun görevi, problemleri (belirli bir çözümün kalitesini temsil eden) bazı değerlendirme fonksiyonları açısından modellemek ve ardından bu amaç fonksiyonunu en aza indirmek (veya soruna bağlı olarak en üst düzeye çıkarmak) için bir arama algoritması kullanmaktır (Burke ve Kendall, 2014). Bilinen pek çok optimizasyon yöntemi vardır. Ancak bu yöntemlerin çoğu belirli bir problemin çözümü için geliştirilmiştir. Bu nedenle optimizasyon probleminin çözümü için gerekli olan yöntemlerin bulunması için optimizasyon probleminin türünün belirlenmesi büyük önem taşır. Bu anlamda optimizasyon yöntemlerinin problemin karar değişkenleri, ele alınan dönem, problemdeki amaç sayısı gibi farklı karakteristikler açısından gruplara ayrılması mümkün olur (Deb, 2014).

### 2.3.1 Tek Amaçlı Optimizasyon

Optimizasyon problemleri amaçlarına göre, tek amaçlı ve çok amaçlı optimizasyon yöntemleri olmak üzere ikiye ayrılır. Tek amaçlı bir fonksiyonda öznitelik sayılarını ve sınıflandırma doğruluğunu toplama yöntemi, tek amaçlı bir optimizasyon yöntemi olarak kabul edilir (Kaya ve Fiğlali, 2016). Elde edilen bir çözümün diğer çözümlere üstünlüğü, tek amaçlı problemlerde bu çözümdeki amaç fonksiyonlarının değerleri kontrol edilerek sağlanır (Al-Tashi ve diğ., 2020).

### 2.3.2 Çok Amaçlı Optimizasyon

Çok amaçlı optimizasyon yaklaşımında, performans kriterlerinde birden fazla amacın sistematik ve eş zamanlı optimizasyonu söz konusudur. Elde edilen bir çözümün diğer çözümlere üstünlüğü, çok amaçlı problemlerde en iyi çözümü elde etmek için kullanılır (Al-Tashi ve diğ., 2020). Araştırmacılar, çok amaçlı problemlerin çözümünde etkin sonuçlara

ulaşmak için çözüm uzayının tamamını kapsayan yöntemler geliştirmiş ve bunları çözüm algoritmalarına uyarlamıştır (Kaya ve Fiğlali, 2016).

Öznitelik alt kümelerinin önemi, farklı yönlerden ele alınan birden fazla kriterin optimizasyonu ile tanımlanabilir. Ancak hedefler çatıştığında bu karmaşık olabilir. Bu karmaşıklığın üstesinden gelmek için çok amaçlı optimizasyon stratejileri kullanılabilir. Çok amaçlı optimizasyon problemleri, çoklu çatışmaları optimize etmek için amaç fonksiyonları içerir (Al-Tashi ve diğ., 2020).

Minimizasyon için çok amaçlı bir optimizasyon problemi matematiksel olarak şu şekilde ifade edilebilir:

$$\text{Min} \qquad F(x) = (f_1(x), f_2(x), \dots, f_n(x)) \tag{2.8}$$

$$\text{Hedef} \qquad g_i(x) \leq 0\,, \qquad i = 1,2, \dots, m \tag{2.9}$$

$$h_i(x) \leq 0\,, \qquad i = 1,2, \dots, l \tag{2.10}$$

Burada, Denklem (2.8) çözüm alanı, Denklem (2.9) amaç fonksiyonu vektörü, Denklem (2.10) ise eşitsizlik kısıtlarını ifade etmektedir. Ayrıca, $\Omega$ karar uzayı olmak üzere, $x \in \Omega$ ve $x = (x_1, x_2, \dots, x_n) \in \Omega \subset R^n$ uygulanabilir bir karar değişkenleri kümesidir. $f: R^n \to R^n$ olup, $R^n$ uzayına değişken uzay denir. $n$ amaç sayısı ve $f_i\ (i = 1,2, \dots, n)$ fonksiyonları amaç fonksiyonlarıdır. Ayrıca $k-$boyutlu $R^n$ uzayı, kriter uzayı olarak da bilinir (Thiele ve diğ., 2009).

Çok amaçlı optimizasyon aşağıdaki özelliklere sahiptir (Deb 2014, syf. 405):

- Optimum kümenin kardinalitesi genellikle birden fazladır,
- Optimizasyonun bir yerine iki farklı hedefi vardır,
- İki farklı arama uzayına sahiptirler.

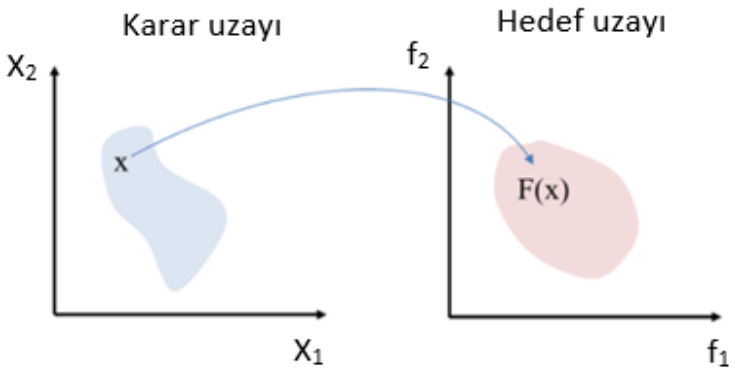
Tek amaçlı optimizasyon ile çok amaçlı optimizasyon arasındaki temel fark, optimal kümedeki kardinaliteden kaynaklansa da, optimizasyon problemi ister tek amaçlı ister çok amaçlı olsun, pratikte bir kullanıcının ihtiyaç duyduğu tek bir çözüm vardır. Çok amaçlı optimizasyonda, kullanıcı en uygun çözümlerden hangisini seçeceği konusunda bir ikilem içindedir. Buna bir cevap aramak için, ideal olarak tüm hedeflerin önemli olduğunu göz önünde bulundurarak,

optimal bir çözüm kümesi bulunmalıdır. Kullanıcı daha sonra bir seçim yapmak için daha yüksek düzeyli kalitatif değerlendirmeler kullanabilir. Bu nedenle, ideal çok amaçlı optimizasyon prosedürü olarak aşağıdaki prosedür önerilir (Deb 2014, syf. 407):

**Adım 1** Hedefler için geniş bir değer yelpazesiyle birden çok değiş tokuşla en uygun çözümler bulunur.

**Adım 2** Daha üst düzey bilgileri kullanarak elde edilen çözümlerden biri seçilir.

Çok amaçlı optimizasyonda, amaç fonksiyonları kullanılarak karar uzaylarının amaçlar uzayına eşlenmesi söz konusudur. Ana ilgi nesnel uzaydadır. Optimal karar seçeneklerini belirleyebilmek için öncelikle çözümler karşılaştırılır daha sonra amaç uzayındaki sıra ilişkisine göre bir sıralama yapılır. Objektif uzaydaki sıra ilişkileri, Pareto baskınlığının en yaygın olarak kullanıldığı baskınlık kavramı kullanılarak tanımlanabilir. Çok amaçlı optimizasyonda Şekil 2.4'deki gibi verilen $x_1$ ve $x_2$ çözümlerini karşılaştırmak çok amaçlı problemlerde tek amaçlı problemlerde olduğu kadar kolay değildir. Bazen hedeflerde çelişkiler ortaya çıkar ve çelişen hedeflerin farklı algoritmalar kullanılarak işlenmesi ve çeşitli takas yöntemleri kullanılarak dengelenmesi gerekir (Afshari ve diğ., 2019).




**Şekil 2.4:** Karar uzayından hedef uzayına tanımlanan bir durumun gösterimi (Feng ve diğ., 2022).

Çok amaçlı optimizasyonda, optimizasyon sonuçlarını belirlemede önemli olan bileşenlerden birisi algoritmalardır. Çok amaçlı optimizasyon, kullanılan algoritmalar bakımından analitik ve sayısal yöntemler olmak üzere iki açıdan ele alınabilir. Analitik yöntemlerde katı matematiksel ispatlar kullanılır ve kesin bir optimizasyon kararı verilirken, sayısal yöntemlerde çeşitli algoritmalar söz konusu olup tüm gerçek dünya problemleri için geniş bir uygulama yelpazesi vardır (Cui ve diğ., 2017).

Bu tez çalışmasında kullanılan Pareto Optimal ve NSGA-II yaklaşımları 2.3.2.1 ve 2.3.2.2 bölümlerinde anlatılmıştır.

#### *2.3.2.1 Pareto Optimal*

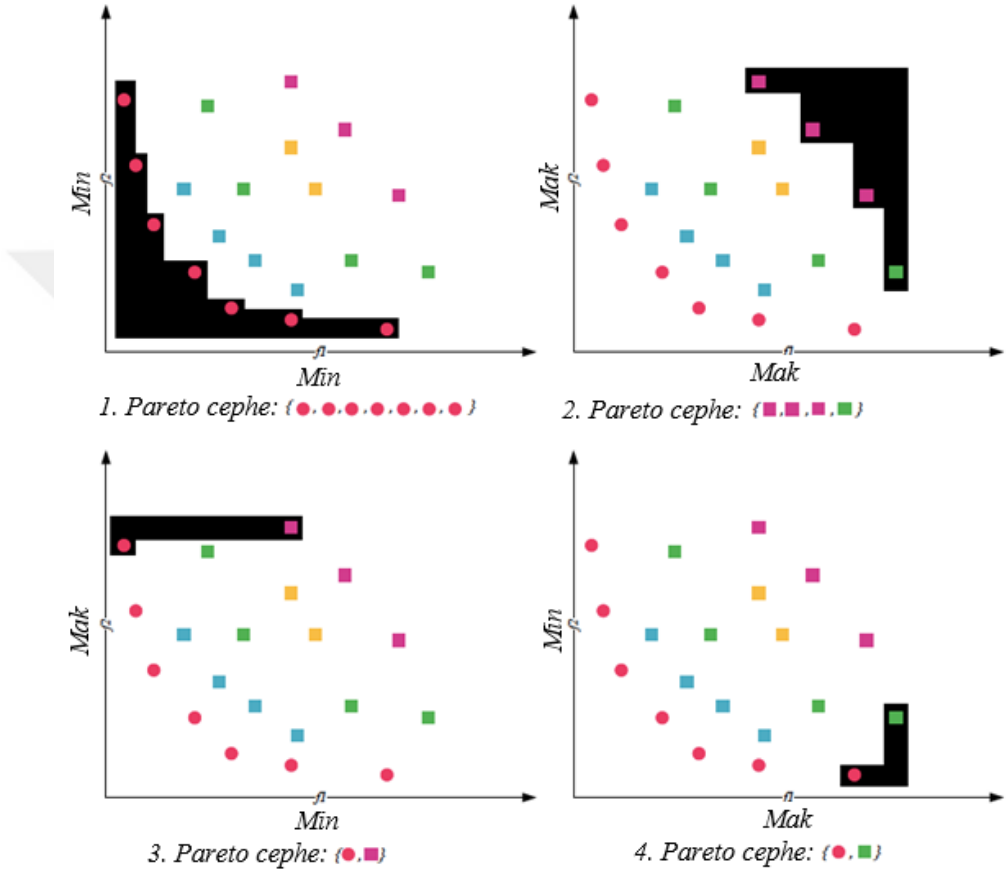
Birçok gerçek dünya arama ve optimizasyon problemi, problemin doğası gereği birden fazla amacı olan doğrusal olmayan programlama problemleri olarak ortaya çıkar. Ancak uygun çözümlerin olmaması nedeniyle bu tür problemler tek amaçlı problemlere dönüştürülerek çözüm elde edilir. Bu dönüşüm aşamasında esas zorluk tek bir optimum çözüm yerine bir dizi optimal çözümün var olmasından kaynaklanmaktadır. Burada hiçbir amaç fonksiyonunun diğerlerinden daha iyi olmadığı düşünülebilir ve tüm amaç fonksiyonlarını eş zamanlı olarak optimize eden tek bir çözümün varlığı konusunda çelişki ortaya çıkar. Bu sebeple optimal çözümler kümesi söz konusu olur (Burke ve Kendal, 2014).

Maliyet ve fayda şeklinde iki amaca sahip bir optimizasyon problemi söz konusu olduğunda, bu karar verme probleminde tek amaç maliyetin minimumda kalması olursa fayda göz ardı edilir. Benzer şekilde tek amaç maksimum fayda sağlanması olursa maliyet göz ardı edilir. Bazı istisnai durumlar dışında minimum maliyete sahip bir çözümün maksimum fayda sağlaması beklenmez. Bu optimizasyon probleminin iki uç çözümünün iki bağımsız optimizasyon problemi olarak değerlendirilmesine gerek yoktur. Bu durumda bu iki uç çözüm arasında maliyet ve fayda açısından bir tercihin söz konusu olduğu başka bir çözüm mevcuttur. Herhangi iki çözüm arasında, biri bir amaç açısından daha iyidir, ancak bu iyileştirme yalnızca diğer amaç için bir fedakarlıktan gelir. Literatürde bu tür çözümler çoğu zaman değiş –tokuş ya da takas çözümleri olarak adlandırılır. Çoğu zaman bu değiş – tokuş çözümleri nesnel değerlerle çizilen nesnel bir uzayda açık bir cephe sağlar. Bu cepheye Pareto-optimal cephe, bu çözümlerin hepsine de Pareto-optimal çözümler adı verilir (Deb, 2014).

Öznitelik seçimi bağlamında bu kavramları yeniden ele alalım. Denklem (2.8)'deki şekilde tanımlanan birçok amaçlı bir optimizasyon probleminde, $y, z \in \Omega$ iki uygun çözüm olsun. $k \in [1, n]$ olmak üzere,

- Her $k$ değeri için $f_i(y) \leq f_i(z)$ yani $y$ çözümü tüm hedeflerde $z$ çözümünden daha kötü değildir,
- En az bir $k$ değeri için $f_i(y) < f_i(z)$ koşulu sağlanıyorsa yani $y$ çözümü en az bir amaçta $z$ çözümünden kesinlikle daha iyidir,

koşullarından en az birisi sağlanıyorsa $y \prec z$ dir yani $y$ çözümü $z$ çözümü üzerinde pareto baskındır. $x^* \neq x \in \Omega$ için eğer $x$ üzerinde Pareto baskınlığı olan başka bir $x^*$ çözümü yoksa $x^* \in \Omega$ Pareto optimal çözüm olarak tanımlanır. Pareto optimal çözümlerin kümesi Pareto optimal kümeyi oluşturur. Pareto kümesinin hedef uzayına eşleştirilmesi ile elde edilen cephe ise Pareto cephe olarak bilinir (Ni ve diğ., 2019; Dong ve diğ., 2021).




**Şekil 2.5:** Pareto cepheler (Deb ve diğ., 2002).

Şekil 2.5 üzerinde siyah taralı alanlar ile iki amaçlı bir problem için Pareto optimal cepheler gösterilmiştir. Burada iki hedefin en aza indirilebileceği veya en üst düzeye çıkarılabileceği dört olası durum dikkate alınmıştır. Her durumda, siyah bölgelerde vurgulanan çözümler, her iki hedefte de eşit amaç değeri gösterdikleri ve en az birinde daha iyi oldukları için beyaz alanlardaki diğer çözümlere baskın çıkar.

### *2.3.2.2 Baskın Olmayan Sıralı Genetik Algoritma (NSGA-II)*

Evrimsel algoritmalar, doğadaki evrim sürecini taklit eder. Bilinen en saygın evrimsel algoritmalardan biri, Darwin'in evrim teorisinden ilham alan genetik algoritmadır. Genetik

algoritma (GA), doğal organizmaların doğal seçilim kullanarak evrimleştiği teorisine dayanmaktadır. GA'da seçim operatörü, çözümlerin kalitesine karar vermek için bir kritere ihtiyaç duyar, bu nedenle çok amaçlı problemlerin çözümü için yetersizdir. Bu nedenle GA, çok amaçlı problemleri çözmek için modifikasyon gerektirir (Deb ve diğ., 2002).

Literatürde çok amaçlı genetik algoritmaların çeşitli versiyonları bulunmaktadır. Bu algoritmalardan biri de 1994 yılında Srinivas ve Deb tarafından geliştirilen baskın olmayan sıralı genetik algoritmadır (*Non-dominated Sorting Genetic Algorithms* - NSGA) (Srinivas ve Deb, 1994). Bununla birlikte, NSGA algoritması, baskın olmayan sıralamanın yüksek hesaplama karmaşıklığı, elitizm eksikliği, paylaşım parametresini belirleme ihtiyacı gibi sorunlar nedeniyle literatürde uzun yıllardır eleştirilmiştir (Deb ve diğ., 2002).

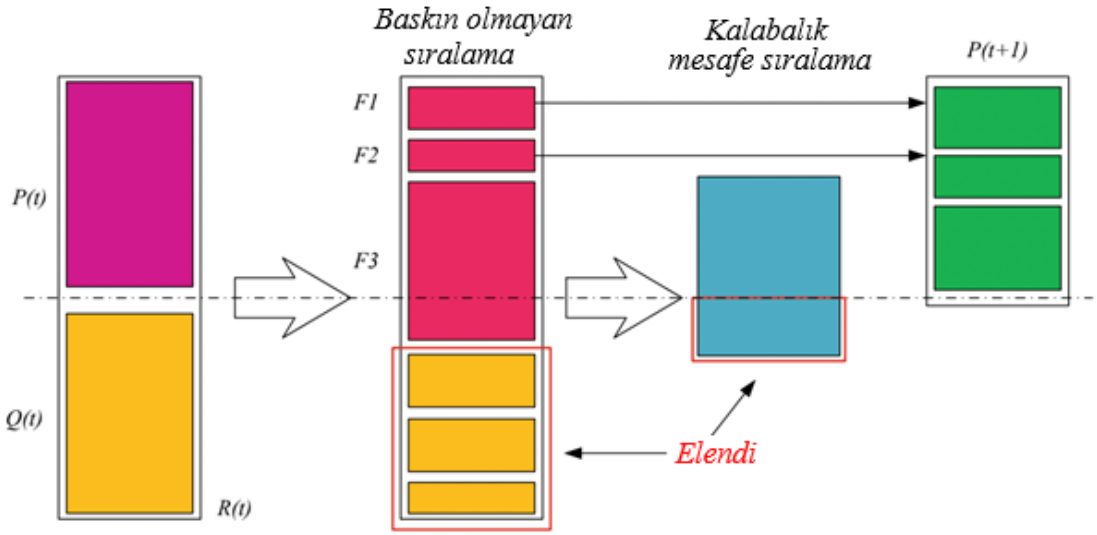
NSGA-II algoritması ise, 2002 yılında Deb ve diğerleri tarafından önerilmiştir ve NSGA'nın geliştirilmiş bir versiyonudur (Deb ve diğ., 2002). Hem NSGA hem de NSGA-II yönteminde, GA, doğası gereği birden fazla hedefi dikkate alan baskınlık düzeylerine göre bireyleri sıralayacak şekilde değiştirilir. NSGA-II, hesaplama maliyeti ve elitizm açısından NSGA'dan daha iyidir (Deb ve diğ., 2002).

NSGA-II, yeni bir parametresiz nişleme operatörü, elitist bir tutma yaklaşımı ve hızlı, baskın olmayan bir sıralama tekniği kullanır. Sonraki paragraflarda bu bileşenler için açıklamalar yer almaktadır (Deb ve diğ., 2002):

Hızlı baskın olmayan sıralama: NSGA-II'nin baskın olmayan türü, N'nin bir popülasyondaki bireylerin sayısı ve M'nin amaç işlevi olduğu $O(MN^3)$ düzenindedir. Bu sıralama, baskın olmayan düzeylerde $(MN^2*N)$ sıralama ve tüm amaç fonksiyonlarına dayalı olarak tüm parçacıkları $(MN^2)$ ile karşılaştırmanın sonucudur. Bu hesaplama maliyetini düşürmek için Mirjalili ve Dong (2020) tarafından önerilen yeni yöntemde baskın olmayan seviyelerin hiyerarşik bir modeli önerilmiştir. Bu nedenle, ilk baskın olmayan seviyeden $(O(MN^2))$ sonra, tüm baskın bireyleri karşılaştırmaya gerek yoktur (Mirjalili ve Dong, 2020).

Elitist tutma yaklaşımı: Mevcut popülasyonun daha önce keşfedilen en iyi baskın olmayan çözümlerle karşılaştırılması, otomatik olarak elitizm ile sonuçlanır. GA'da, mevcut popülasyonun bir yüzdesinin korunduğu ve bir sonraki nesile bozulmadan aktarıldığı bir elitizm olasılığı vardır. Çok amaçlı problemlerle uğraşırken mevcut birkaç en iyi çözüm olduğu gerçeğinden dolayı, bu, NSGA-II de dahil olmak üzere çok amaçlı GA'lara dahil edilmemiştir.

Bununla birlikte, belirli NSGA varyasyonları elitizm kullanır (Deb ve Goel, 1993; Mirjalili ve Dong, 2020).




**Şekil 2.6:** NSGA-II Prosedürü (Deb ve diğ., 2002).

Şekil 2.6, NSGA-II prosedürüne ait çalışma sürecini göstermektedir. Bu şekil, NSGA-II'nin baskın olmayan bir sıralama yaklaşımı kullanarak bireyleri nasıl sıraladığını ve derecelendirdiğini göstermektedir. Bunu takiben, seçim operatörü yeniden üretimde yer alma şanslarına bağlı olarak bunlardan herhangi birini seçmektedir (Mirjalili ve Dong, 2020).

Yeni nişleme operatörü (kalabalık mesafesi): En yakın komşu yoğunluğu, bir küp veya hiperküp olabilen en yakın komşulardan oluşan bir bölgenin çevresini tahmin eder. Bu kriterlere göre daha yüksek değerlere sahip bireyler lider olarak seçilir. Liderler, bu ölçütlerde hangi kişilerin daha büyük değerlere sahip olduğuna göre seçilir (Kukkonen ve Deb, 2006; Mirjalili ve Dong, 2020).

Niş tekniğinde bir çap ($\sigma_{share}$) tanımlanmalıdır ve sonuçlar büyük ölçüde buna bağlıdır. Ancak, önerilen operatörde tanımlanacak bir parametre yoktur. Bir çapın (veya "payın (share)") tanımı, nişleme tekniğinin sonuçları üzerinde önemli bir etkiye sahiptir. Bununla birlikte, önerilen operatör belirtilen bir parametreye sahip değildir.

Python programlama dilinde NSGA-II algoritmasını jMetalPy (Benítez-Hidalgo ve diğ., 2019) ve Pymoo (Blank ve Deb, 2020) kütüphaneleri ile kullanmak mümkündür.

## 2.4 LİTERATÜR TARAMASI

Bu tez çalışması kapsamında Web of Science, Google Scholar ve Ulusal Tez Merkezi üzerinden literatür taraması gerçekleştirilmiştir.

Web of Science veri tabanında yapılan sorgulamalar Tablo 2.1'de detaylı olarak belirtilmiştir. Yapılan dört farklı sorgu sonrasında elde edilen toplam 157 adet makale detaylı olarak incelenmiş, tez konusu ile paralel olan çalışmaların detaylı incelemesi yapılmıştır.

**Tablo 2.1:** Web of Science literatür taraması sorguları.

| Sorgu | Periyot | Sonuç |
|---|---|---|
| (((ALL= (feature selection)) AND ALL = (correlation)) AND ALL=(consistency)) AND ALL = (Pareto) | 1975-2022 | 1 |
| (((ALL=(feature selection)) AND ALL=(correlation)) AND ALL=(consistency)) AND ALL=(NSGA) | 1975-2022 | 0 |
| ((ALL=(multi-objective feature selection)) AND ALL=(consistency)) AND ALL=(correlation) | 1975-2022 | 2 |
| ((ALL=(feature selection)) AND ALL=(consistency)) AND ALL=(correlation) | 2018-2022 | 154 |

Xu ve diğ. (2018) çalışmasında mevcut öznitelik seçimi yaklaşımlarının eksik ön bilgi ile etiket korelasyon bilgisini çıkarmak ve bu bilgiyi öznitelik seçimine yerleştirmek için birleşik bir çerçeveyi nadiren ele aldığı varsayımıyla, bir tutarlılık bazlı öznitelik seçimi modeli önermişlerdir ve bu sayede seçim çalışmasının tutarlılığını ölçmeye çalışmışlardır. Sonuçta gerçek dünya verilerinin kapsamlı bir deneysel değerlendirmesi sayesinde önerilen yaklaşımın çeşitli değerlendirme ölçütleri altında üstün olduğunu göstermişlerdir.

Dranca ve diğ. (2018) çalışmasında hastalığın farklı aşamasında olan Parkinson hastalığına sahip olan 30 hasta ile klinik bir çalışma yapmışlardır. Bu tez çalışmasında hasta bulgularının kaydedilmesi için Kinect kullanmışlar ve elde edilen veriler için yapılan ön işleme aşamaları

sonrasında 115 öznitelik tanımlanmış ve bu öznitelikler içerisinden seçim yapabilmek için korelasyona dayalı öznitelik seçimi, bilgi kazanımı ve CONS öznitelik seçim yöntemleri ile karar ağaçları, Bayesian ağlar, sinir ağları ve K – en yakın komşuluk sınıflandırıcıları gibi farklı sınıflandırma yöntemlerini kullanmışlardır. En iyi sonuç veren sınıflandırma çalışması, korelasyona dayalı öznitelik seçimi ile seçilen yedi özniteliğe sahip öznitelik alt kümesi üzerinde Bayesian ağlar sınıflandırıcısı ile elde edilmiştir.

Abu Shanab ve Khoshgoftaar (2018) çalışmasında, gürültü nedeniyle değişen öğrenme güçlüğü seviyelerine sahip olan yüksek boyutlu gerçek biyoinformatik veri kümelerini kullanan iki filtre tabanlı alt küme seçim tekniğini karşılaştırmıştır. Bu amaçla yaygın olarak kullanılan dört sınıflandırıcı kullanarak sınıflandırmayı yaptıktan sonra korelasyon bazlı öznitelik seçim yöntemi (CFS) ve tutarlılık bazlı öznitelik seçim yöntemini (CONS) kullanmışlardır. Sonuç olarak ise CFS yönteminin etkili bir şekilde tüm zorluk seviyelerinde CONS yönteminden daha iyi performans gösterdiği elde edilmiş ve yalnızca önemli genleri tanımlamakla kalmayan, aynı zamanda gereksiz genleri de ortadan kaldıran alt küme değerlendirme tekniği olarak CFS kullanılması önerilmiştir.

Z.-Z. Liu ve diğ. (2018) çalışmasında, ilaç geliştiricileri için öznitelik seçimi için evrimsel hesaplamayı kullanan ve Github'da yayınlanan bir yazılım geliştirmiştir. Yazılım, regresyon ve sınıflandırma için tek ve çok amaçlı evrimsel algoritmalar kullanmaktadır.

Al-Tashi ve diğ. (2019) çalışmasında, öznitelik seçim problemlerini çözmek için hibrit gri kurt optimizasyonunun ve parçacık sürü optimizasyonunun ikili bir versiyonunu önermiştir.

Azarmdel ve diğ. (2020) çalışmasında dut meyvesini olgunluklarına göre (olgunlaşmamış, olgun ve fazla olgun) analiz etmek için görüntü işleme ve yapay zeka tekniklerini kullanmıştır. Toplamda 577 dut meyvesi uzmanlar tarafından incelenerek derecelendirilmiş ve görüntüleri bir sisteme aktarılmıştır. Daha sonra her bir parçalı dut meyvesinin geometrik özellik, renk ve doku özellikleri korelasyon bazlı öznitelik seçim alt kümesi (CFS) ve tutarlılık alt kümesi (CONS) olmak üzere iki öznitelik indirgeme (*feature reduction*) yöntemi kullanılarak çıkarılmıştır. Dut meyvesini sınıflandırmak için ise yapay sinir ağları ve destek vektör makinesi yöntemleri kullanılmıştır. Yapay sinir ağları ve destek vektör makineleri ile birleştirilen yapay görme sistemi dutları olgunluk seviyelerine göre başarılı bir şekilde sınıflandırmıştır.

Hu ve diğ. (2020) çalışmasında literatürde var olan çalışmaların veri bilgisini ya öznitelik matrisi ya da etiket matrisi şeklinde iki veri matrisi ile tanımladığına dikkat çekerek, öznitelikler ve etiketler arasında paylaşılan ortak mod dikkate alınarak öznitelik seçimi (*feature selection considering shared common mode* - SCMFS) adlı yeni bir öznitelik seçimi metodu önermişlerdir. Consistency kavramını, seçim öncesi ve sonrası veri kümeleri arasındaki tutarlılık için kullanılmışlardır. Önerilen metot için on beş farklı gerçek veri kümesi üzerinde uygulamalar yaparak bu yöntemin sınıflandırma üstünlüğü olduğunu elde etmişlerdir.

Algin ve diğ. (2020) çalışmasında göçmen kuşlar (*migrating birds*) optimizasyonu, benzetilmiş tavlama (*simulated annealing*), diferansiyel evrim (*differential evolution*), ve parçacık sürü optimizasyonu tekniklerini UCI veri deposundan alınan iyi bilinen on yedi veri kümesine uygulamıştır. Bu durum, özellikle göçmen kuşlar optimizasyonunun öznitelik seçimi problemlerinin çözümündeki ilk uygulamasıdır. Arama algoritmaları tarafından oluşturulan alt kümelerin kalitesini değerlendirmek için bu tez çalışmasında olasılıksal (*probabilistic*) tutarlılık bazlı öznitelik seçimi ve korelasyon bazlı öznitelik seçimi olmak üzere iki alt küme değerlendirme yöntemi uygulanmıştır. Algoritmaların performans karşılaştırması K – en yakın komşu, naive bayes ve C4.5 karar ağacı sınıflandırıcıları kullanılarak yapılmıştır. Sonuçta göçmen kuşlar optimizasyonu tabanlı filtreleyici yaklaşımın diğer üç yönteme kıyasla en iyi performansı gösterdiği elde edilmiştir. Ayrıca, bir alt küme değerlendiricisi olarak korelasyon bazlı öznitelik seçim algoritmasının, olasılığa dayalı tutarlılık bazlı öznitelik seçim algoritmasından daha iyi performans gösterdiği ve bir sınıflandırıcı olarak C4.5'in K – en yakın komşu ve naive bayes ile karşılaştırıldığında daha iyi sonuçlar aldığı gözlemlenmiştir.

Jiarpakdee ve diğ. (2020) çalışmasında yaygın olarak kullanılan 11 öznitelik seçimi algoritması ve çalışmada önerilen AutoSpearman tekniğini üretilen metriklerin tutarlılığı, üretilen metrik alt kümelerinin korelasyonu, performans, hesaplama maliyeti ve yorumlama üzerindeki etkisi şeklinde beş boyutta araştırmışlardır. Çalışmada halka açık 14 farklı veri seti söz konusu beş boyutta incelenmiştir. AutoSpearman hariç diğer öznitelik seçimi yöntemlerinin genellikle tutarsız ve ilişkili olduğu bulunmuştur. Bu çalışma "kusur tahmin" alanında yaygın olarak kullanılan öznitelik seçimi teknikleri tarafından üretilen metrik alt kümelerinin doğası gereği tutarsız (*inconsistent*) olduğunu gösteren ilk çalışmadır.

Usman ve diğ. (2020) çalışmasında kazanç oranı tabanlı entropi, NSGA-II, NSGA-III ve guguk kuşu *(cuckoo*) optimizasyon algoritması ile birlikte karşılıklı bilgi kullanılarak çok amaçlı filtre tabanlı öznitelik seçim yaklaşımları önermiştir.

Al-Tashi ve diğ. (2020) çalışmasında öznitelik seçim problemlerini optimize etmek için sigmoid transfer fonksiyonuna dayalı Çok Amaçlı Gri Kurt iyileştiricinin ikili versiyonunu geliştirmiştir.

Othman ve diğ. (2020), kanser mikrodizi verileri kullanılarak gen seçimi için evrimsel operatörler yardımıyla hibrit çok amaçlı bir guguk kuşu *(cuckoo)* arama yöntemi önermiştir.

El-kenawy ve diğ. (2020), COVID-19'un öznitelik seçimi ve sınıflandırılması için AlexNet adlı konvolüsyonel sinir ağı, stokastik fraktal aramaya dayalı kılavuzlu balina optimizasyon algoritması ve parçacık sürü optimizasyonuna dayalı kılavuzlu balina algoritması kullanan iki optimizasyon algoritması önermiştir.

Raj ve diğ. (2020) çalışmasında hastalık veri kümelerinin sınıflandırılması için derin öğrenme ve derin öğrenme sınıflandırıcısının performansını artırmak için muhalif tabanlı karga arama algoritması önermiştir.

Elhariri ve diğ. (2020) çalışmasında tarihi yapılardaki en önemli kusurlardan birisinin şiddetine göre potansiyel yapısal bozulmanın bir göstergesi olan yapısal çatlaklar olduğunu vurgulayarak, çatlak şiddeti tanıma sisteminde salp sürü optimizasyonu kullanarak çok amaçlı öznitelik seçim yöntemi ile hibrit filtreleyici-sarmalayıcı optimizasyon kullanan yeni bir çatlak şiddeti tanıma sistemi sunmak için bir öznitelik seçim yöntemi önermiştir.

Zhang ve diğ. (2021) çalışmasında, hata tanımanın etkinliğini artırmak ve hesaplamanın karmaşıklığını azaltmak için gizli bayes sınıflandırıcısı *(hidden bayesian classifier)* ile birleştirilmiş filtre yöntemlerine dayalı bir öznitelik seçme yöntemi önermektir. Çelik şerit yüzey kusurlarının doğru bir şekilde sınıflandırılabilmesi için en uygun hibrit modeli seçmek amacıyla ilk olarak büyük bir öznitelik kümesi elde edildi. Daha sonra öznitelik uzayını optimize etmek için üç öznitelik seçimi yöntemi (korelasyon bazlı öznitelik seçimi, tutarlılık bazlı alt küme skorlama ve bilgi kazanımı yöntemleri) kullanıldı. Öznitelik seçimi yöntemleri için parametreler gizli naive bayes algoritmasının sınıflandırma doğruluğu sonuçlarına dayanmaktadır. Sonuçta, öznitelik seçimi yaklaşımlarıyla birleştirilen gizli naive bayes

yönteminin diğer hata tanıma modellerine kıyasla daha iyi sınıflandırma performansına sahip olduğu ve önerilen hibrit modelin en sağlam, en etkili ve en yüksek sınıflandırma doğruluğuna sahip model olduğu elde edilmiştir.

Aljarah ve diğ. (2021) çalışmasında öznitelik seçme görevleriyle başa çıkmak için çoklu evren iyileştiricinin ikili çok amaçlı kullanımını önermiştir. Önerilen yaklaşım, minimum sınıflandırma hatası oranını korurken ilgisiz ve/veya gereksiz öznitelikleri etkili bir şekilde ortadan kaldırabildiğini göstermektedir.

Assafo ve Langendoerfer (2021) çalışmasında çok amaçlı duyusal özniteliklerin seçimi için TOPSIS'in ve kendi kendini düzenleyen haritaya dayalı anormallik saptama yaklaşımının kullanıldığı bir öznitelik seçimi yaklaşımı önermiştir.

El-Kenawy ve diğ. (2021) çalışmasında göğüs röntgeni görüntüleri üzerinde, gelişmiş sincap arama optimizasyon algoritmasına dayalı olarak konvolüsyonel sinir ağı ve çok katmanlı algılayıcı sinir ağı'nın kullanıldığı iki aşamalı bir yöntem önermiştir.

Alia ve Taweel (2021), ikili guguk kuşu arama *(binary cuckoo search*) ve kaba küme teorisine dayalı yeni bir filtreleyici öznitelik seçimi yaklaşımı önermiştir.

Agrawal ve diğ. (2021), öznitelik seçimi için metasezgisel algoritmaları dört kategoride sınıflandırmış ve yüzden fazla metasezgisel algoritmanın kategorik bir listesini sunmuştur.

Albashah ve Rais (2022) çalışmasında, popülasyon faktörü kullanılarak başlatma popülasyonunun etkisi araştırmıştır. Öznitelik seçimi için ikili çok amaçlı gri kurt optimizasyonunda popülasyon başlatma yöntemiyle elde edilen yirmi kurt ve öznitelik sayısı ve hata oranının minimize edilmesi amaçlanmıştır. Önerilen yöntem, petrol ve gaz veri kümeleri kullanılarak önceki çalışma olan ikili çok amaçlı gri kurt optimizasyon algoritması ile karşılaştırılmıştır. Önerilen yöntem, öznitelik sayılarını ve hata oranlarını azaltma açısından daha iyi performans göstermiştir.

Zhang ve diğ. (2022) çalışmasında öznitelik seçimi için çok amaçlı Harris Hawks optimizasyon algoritması, ilişkisel öğrenme, gri kurt optimizasyonu ve kaotik yerel arama kullanmaktadır.

Ghaleb ve diğ. (2022), Yapay sinir ağları ve Grasshopper optimizasyon algoritması ile spam tespit sistemi için çok amaçlı öznitelik seçim yöntemi önermiştir.

Ghanem ve diğ. (2022) çalışmasında saldırı tespit sistemi için izinsiz giriş tespit sistemi mekanizmaları da dahil olmak üzere daha güvenilir koruma sistemlerinin geliştirilmesi amacıyla çok amaçlı bir BAT algoritması üzerinde sarmalayıcı öznitelik seçimi yaklaşımı kullanılmıştır.

Piri ve diğ. (2022a), çalışmasında COVID-19 hastalarının sağlık durumu tahmininde ilgili öznitelikleri keşfetmek için ikili çok amaçlı hibrit filtre-sarmalayıcı şempanze optimizasyonu (EBMOChOA-FW) tabanlı öznitelik seçimi yaklaşımını kullanmıştır.

Piri ve diğ. (2022b), ABD'de bebek ölümlerine neden olan çeşitli etkili faktörleri çıkarmayı ve anlamayı amaçlamaktadır. Çalışmada, öznitelik seçimi için istatistiksel kanıtlarla birlikte, kalabalık mesafesine dayalı çok amaçlı bir karınca aslanı optimizasyonu önerilmiştir.

Piri ve diğ. (2022c) çalışmasında tıbbi veri kümelerinde ilgili öznitelikleri araştırmak için KNN ve ikili çok amaçlı şempanze optimizasyon algoritmasını (BMOChOA) kullanmıştır. Bu araştırmada, BMOChOA algoritması için grup bilgilerine ve kullanılan kaotik fonksiyon türlerine göre 12 farklı varyasyon elde edilmiştir. Önerilen BMOChOA varyasyonlarından elde edilen en iyi Pareto cephesi, değişken boyutlarda 14 popüler tıbbi veri seti alınarak üç karşılaştırmalı çok amaçlı öznitelik seçim yöntemiyle karşılaştırılmıştır. Daha sonra deneysel çıktıları dört çok amaçlı performans değerlendirici yöntemi kullanarak analiz etmişler ve önerilen öznitelik seçim yönteminin öznitelik sayısı ve sınıflandırma performansı açısından üstün olduğunu bulmuşlardır.

Jimenez ve diğ. (2022) çalışmasında halka açık veri kümelerini kullanarak kanser geni ifadesi *(gene expression)* ve genotip-doku ifadesi sınıflandırma görevleri için uygulanan ikili korelasyon ve ikili tutarlılığa dayalı iki yeni çok değişkenli özellik sıralama yöntemi önermiştir. Önerilen yöntemlerin, çok amaçlı evrimsel arama stratejisi ve gömülü öznitelik seçim yöntemleri C4.5 ve LASSO ile korelasyon ve tutarlılığa dayalı öznitelik alt kümesi değerlendirmesi için diğer özellik seçme yöntemlerinin yanı sıra kümeleme varyasyonu *(clustering variation),* ki kare, korelasyon, bilgi kazanımı, ReliefF ve anlamlılık *(significance)* gibi en gelişmiş öznitelik sıralama yöntemlerinden daha iyi performans gösterdiğini istatistiksel olarak kanıtlamışlardır.

Yükseköğretim Kurulu Başkanlığı Ulusal Tez Merkezi web sitesinde yapılan taramaya göre, periyot sınırlaması olmadan “Multi-objective feature selection” anahtar sözcüğü için yapılan

taramada bir kaynağa, 2018-2022 yılları arasında “Feature selection” anahtar sözcüğü için 60 sonuca ulaşılmıştır. Periyot sınırlaması olmadan “Multi-objective feature subset selection” anahtar kelimesi için yapılan aramada hiç sonuç dönmemiştir.

Taramadan dönen toplam 61 sonuca ait çalışma detaylı olarak incelenmiştir. Yapılan incelemeler sonrası, bu tez çalışması ile paralellik gösteren dokuz çalışmaya ait detaylar aşağıda belirtilmiştir.

Özmen (2016), “Çok amaçlı değişken seçimine etkileşimli evrimsel yaklaşımlar” başlıklı tez çalışmasında, doğruluk, kardinalite, maliyet ve risk amaç fonksiyonları üzerinden yaklaşım geliştirme çalışılmıştır.

Akbulut (2019), “Sınıflandırma ve özellik seçimi için örtüşebilen hiper kutu kullanan optimizasyon yaklaşımları” isimli doktora tez çalışmasında, karma tamsayılı programlama ve hiper kutu kırpma yaklaşımlarını kullanan bir yaklaşım önermiştir.

Kalaycı (2019), “Metasezgisellerin özellik seçimi problemi için uygulanması” başlıklı yüksek lisans tezinde göçmen kuşlar optimizasyonu, parçacık sürü optimizasyonu, benzetilmiş tavlama, diferansiyel evrim yaklaşımlarını kullanmıştır.

Subaş (2019), “Sürekli/ikili parçacık sürü optimizasyonu ve destek vektör makinelerinin hibrit kullanımı ile özellik seçimi” isimli yüksek lisans tez çalışmasında aday ilaç moleküllerini belirlemekte etkin özelliklerin seçimi için tıp, finans, mühendislik gibi birçok alanda başarılı sonuçlar veren sezgisel algoritmalardan olan sürekli/ikili parçacık sürü optimizasyonu ile destek vektör makineleri için hibrit bir uygulama yapmıştır.

Ezirmik (2020), “Meta-Sezgisel yöntemler ile müzik verisi üzerinde özellik seçimi ve kategorizasyon” isimli yüksek lisans tez çalışmasında, karınca koloni algoritması, parçacık sürü optimizasyonu, benzetilmiş tavlama ve genetik algoritma yöntemlerini kullanmıştır.

Khalafallah (2021), “Sınıflandırma algoritmalarında öznitelik seçimi için evrimsel algoritmaların kullanımı” adlı yüksek lisans tez çalışmasında korelasyon bazlı değişken alt kümesi değerlendirme ve dört farklı evrimsel algoritma kullanmıştır.

Kaygusuz (2021), “Türkçe metinlerin sınıflandırılmasında kullanılmak için geliştirilen melez bir öznitelik seçim yöntemi” başlıklı yüksek lisans tezinde, literatürde bulunan farklı öznitelik skorlama yöntemlerinin birlikte kullanımı üzerine çalışmıştır.

Geren (2022), “Eşzamanlı öznitelik seçimi ve kümeleme için hibrit bir sürü zekası algoritması” isimli yüksek lisans tez çalışmasında, Karınca Kolonisi Optimizasyonu ve Değişken Komşuluk Aramasından oluşan bir sürü zekası algoritması önermiştir.

Çelik (2022), “Özellik seçimi için güncel meta sezgisel optimizasyonları kullanılarak yeni ikili optimizasyon algoritması geliştirilmesi” isimli tez çalışmasında, 97 kısıtsız karşılaştırma fonksiyonu, 14 mühendislik tasarım problemi kullanılmış ve parametrik olmayan Wilcoxon's Signed Rank testi kullanılarak bir başarı testi uygulanmıştır. Öznitelik seçiminde kullanılan algoritmaların performansı 17 veri kümesi üzerinde karşılaştırmalı olarak incelenmiştir.

# 3. MALZEME VE YÖNTEM

Bu bölümde, tez çalışmasında kullanılan araçlar ve önerilen yaklaşım detaylı olarak anlatılmıştır.

## 3.1 ÇOK AMAÇLI KORELASYON VE TUTARLILIK BAZLI ÖZNİTELİK SEÇİMİ

Bu tez çalışması kapsamında gerçekleştirilen literatür taraması çalışmalarında, daha önce CFS ve CONS öznitelik alt kümesi değerlendirme yaklaşımlarını aynı anda uygulayan ve bu iki yaklaşıma ait merit değerlerinin oluşturduğu kartezyen düzlemi oyun alanı olarak kullanan bir yaklaşıma rastlanılmamıştır. Tanımlanan bu oyun alanında amaç, Pareto ve NSGA-II optimizasyon yaklaşımlarını kullanarak, önerilen yaklaşımın etkinliğini arttırmaktır. Önerilen bu yaklaşıma, "*Çok Amaçlı Korelasyon ve Tutarlılık Bazlı Öznitelik Alt kümesi Seçimi*" adı verilmiştir. Bu tez çalışmasının ilerleyen kısımlarında ve literatürde pratik kullanım açısından yöntem adı ÇOKOTUÖS olarak kısaltılmıştır. İngilizce literatürde ise, *"Multi-Objective Correlation and Consistency Based Feature Subset Selection"* olarak adlandırılması ve MOCOCOFS olarak kısaltılması uygundur.

Aday öznitelik alt kümesi uzayı S olmak üzere, s ∈ S için;

$$\max \begin{cases} Merit_{CFS}(s) \\ Merit_{CONS}(s) \end{cases} \tag{3.1}$$

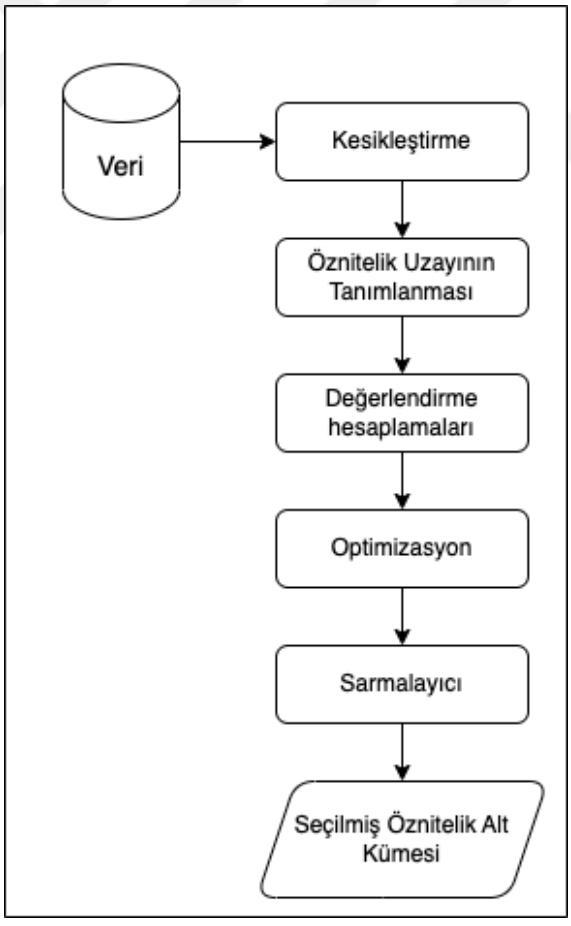
şeklinde CFS ve CONS merit değerlerini aynı anda maksimize etmeye yönelik olarak amaç fonksiyonları tanımlanmıştır. Bu aşamada problem, bir çok amaçlı *(multi-objective)* optimizasyon ve çok kriterli karar verme *(multi-criteria decision making)* problemine dönüşmüştür. Gelinen bu durumda, aynı anda maksimize edilmek istenen $Merit_{CFS}(s)$ ve $Merit_{CONS}(s)$ değerleri için Pareto optimal küme içerisinde yer alan adaylar olası makul çözümler olarak kabul edilebilir. Pareto optimal küme içerisindeki adaylar, birbirlerine bu iki fonksiyonun değerlerine göre bir üstünlük (baskınlık) sağlayamamaktadır. Dolayısıyla bu aşamada, son kararı vermek için eldeki adayları değerlendirmek üzere bir işleme daha ihtiyaç bulunmaktadır.

Öznitelik seçimi çalışmalarında amaçlardan bir tanesi sınıflandırma doğruluğu değerini maksimize etmektir ancak sarmalayıcı yaklaşım ile tüm aday öznitelik alt kümesi için

sınıflandırma modeli oluşturmak ve sonuçları karşılaştırmak oldukça hantal bir yaklaşımdır. Ancak CFS ve CONS merit değerlerine çok amaçlı optimizasyon yaklaşımları uygulayarak adayları filtreleme aşamasında, olası aday sayısı tüm taranan adaylara kıyasla oldukça düşük bir seviyeye indirgendiği için bu aşamada kalan adaylardan hangisinin seçileceğine bir sarmalayıcı yaklaşım ile karar vermek uygun bir çözüm olmaktadır. Sarmalayıcı yaklaşım, uygulanan aday sayısı oldukça düşük düzeyde tutulmakta olduğu için yüksek sınıflandırma doğruluğu elde etmek için makul kabul edilebilecek düzeyde bir işlem yükü gerektirmektedir.

ÇOKOTUÖS yaklaşımı, CFS'yi ortaya atan Hall'un (1999) ifadesine "oldukça tutarlı" ifadesi eklenerek aşağıdaki şekilde ifade edilebilir;

*"İyi bir öznitelik alt kümesi,* ***oldukça tutarlı*** *ve barındırdığı öznitelikler sınıf ile yüksek korelasyona sahip iken kendi aralarında korelasyon bulunmayanlardır."*




**Şekil 3.1:** ÇOKOTUÖS yönteminin akış şeması.

ÇOKOTUÖS yaklaşımında 6 adım yer almaktadır (Şekil 3.1):

1- Kesikleştirme: Bu aşamada veri kümesi içerisinde bulunan sürekli özniteliklerin kategorik hale dönüştürülmesi sağlanır. Bu işlem, CFS ve CONS öznitelik seçimi yaklaşımlarının tanımında bulunmaktadır. Bu aşama için WEKA'da, Fayyad ve Irani'nin kesikleştirme *(discretization)* algoritmasından faydalanılır (Fayyad ve Irani, 1993; Frank ve diğ., 2016).

2- Öznitelik Uzayının Tanımlanması: Bu aşamada skorlanması planlanan öznitelik alt kümelerine ait öznitelik alt kümesi alt uzayı tanımlanır. Bu tez çalışması kapsamında tam arama *(exahustive search)* ve rastgele arama *(random search)* yöntemleri ile belirlenen adaylar üzerinde değerlendirme yapılmaktadır.

3- Değerlendirme Hesaplamaları: Bu aşamada, önceki aşamada belirlenen her bir aday öznitelik alt kümesi için CFS ve CONS merit değerleri hesaplanır ve bir tabloda tüm skorlar saklanır.

4- Optimizasyon: Hesaplanan merit değerlerini maksimize etmek amacıyla çok amaçlı (*multi-objective*) optimizasyon algoritmaları tarafından makul adayların belirlenmesi sağlanır. Bu tez çalışması kapsamında Pareto ve NSGA-II algoritmaları kullanılmaktadır.

5- Sarmalayıcı: Optimizasyon algoritmasının döndürdüğü tüm aday öznitelik alt kümeleri üzerinde sarmalayıcı öznitelik seçimi yaklaşımı uygulanır. Bu aşamadaki tüm adaylar tamamen kategorik özniteliklerden oluştuğu için, kategorik veri kümeleri üzerinde yüksek sınıflandırma doğruluğu vaat eden Catboost (Prokhorenkova ve diğ., 2018) sınıflandırıcı, sarmalayıcı uygulanması için tercih edilmiştir.

6- Seçilmiş Öznitelik alt kümesi: Sarmalayıcı aşamada en yüksek sınıflandırma doğruluğu değeri elde eden öznitelik alt kümesinin sonuç olarak döndürülmesi sağlanır. Aynı sınıflandırma doğruluğu değerine sahip birden çok aday varsa, öznitelik sayısı düşük olan seçilir.

ÇOKOTUÖS yaklaşımının Pareto ve NSGA-II ile birlikte kullanımına dair sözde kod EK-1'de paylaşılmıştır.

CFS ve CONS yaklaşımlarının uygulanması öncesinde bir kesikleştirme aşaması bulunduğu için merit değerlerinin hesaplandığı veri kümeleri tamamen kategorik özniteliklerden oluşmaktadır. Son dönemde kategorik sınıflandırma için önerilen en güncel algoritmalardan bir

tanesi Catboost algoritmasıdır. Bu nedenle, bu aşamada Pareto küme içerisinde yer alan tüm aday öznitelik alt kümeleri için Catboost ile birer sınıflandırma modeli üretilmesi ve en yüksek sınıflandırma doğruluğu değerine sahip öznitelik alt kümesinin seçilmiş öznitelik alt kümesi olarak belirlenmesi önerilmiştir. Eğer Pareto küme içerisindeki adaylardan sınıflandırma doğruluğu değeri aynı olan birden çok aday bulunursa eleman sayısı düşük olanın seçilmesi önerilmiştir. Bu yaklaşımımıza Pareto küme yardımıyla çok amaçlı korelasyon ve tutarlılık bazlı öznitelik seçimi yöntemi *(Multi-Objective Correlation and Consistency based feature selection with Pareto)* (ÇOKOTUÖS-Pareto) adı verilmiştir.

ÇOKOTUÖS-Pareto yaklaşımının en hantal yanının, birden çok aday için sınıflandırma algoritması uygulanan sarmalayıcı aşaması olduğu söylenebilir. Bu aşamayı optimize etmek için, çoğu zaman Pareto kümeden daha küçük bir alternatif çözüm kümesi üreteceği varsayımı ile NSGA-II algoritmasının kullanılması önerilmektedir. Hedef fonksiyonumuz için NSGA-II algoritması çalıştırılarak Pareto kümenin eleman sayısından daha düşük sayıda eleman barındıran alternatif bir çözüm kümesi bulunabilir ve bu sayede sarmalayıcı aşamasında daha az aday değerlendirilerek iyi bir çözüme ulaşılabilir. Bu yaklaşımımıza NSGA II yardımıyla çok amaçlı korelasyon ve tutarlılık bazlı öznitelik seçimi yöntemi *(Multi-Objective Correlation and Consistency based feature selection with NSGA-II)* (ÇOKOTUÖS-NSGAII) adı verilmiştir.

Ayrıca taranan CFS ve CONS merit değerlerinin oluşturduğu kartezyen uzayda her zaman geniş bir Pareto küme elde edilemeyebilir. Pareto yaklaşımı tek bir baskın maksimal değere sahip adayı tespit edebilir. Bu durumda, sarmalayıcı aşamaya gerek kalmaz ve Pareto yaklaşımı ile seçim tamamlanmış kabul edilir. Ancak bu durumda, sarmalayıcı yaklaşımın sınıflandırma doğruluğu değerini yüksek tutma özelliğinden yararlanılamamış olur. Bu tip durumlarda, NSGA-II yaklaşımı ile tespit edilen alternatiflerin üzerinde sarmalayıcı aşamasında sınıflandırma doğruluğu yükseltme amacına yönelik daha çok olası adayın değerlendirilmesi sağlanabilir.

## 3.2 VERİ KÜMELERİ

Bu tez çalışmasında UCI veri deposundan elde edilen 10 adet veri seti kullanılmıştır. İlgili veri setine ait örnek sayısı, öznitelik sayısı ve sınıf sayısı Tablo 3.1'de verilmiştir.

**Tablo 3.1:** Tez çalışmasında kullanılan veri kümeleri.

| | Veri kümesi | Örnek | Öznitelik | Sınıf | Web sitesi |
|---|---|---|---|---|---|
| 1 | Breast Cancer (Michalski ve Chilausky, 1980) | 286 | 9 | 2 | https://archive.ics.uci.edu/ml/datasets/Breast+Cancer |
| 2 | Congressional Voting Records (Schlimmer, 1987) | 435 | 16 | 2 | https://archive.ics.uci.edu/ml/datasets/Congressional+Voting+Records |
| 3 | Primary Tumor (Cestnik ve diğ., 1987) | 339 | 17 | 22 | https://archive.ics.uci.edu/ml/datasets/primary+tumor |
| 4 | Lymphography (Cestnik ve diğ., 1987) | 148 | 18 | 4 | https://archive.ics.uci.edu/ml/datasets/Lymphography |
| 5 | Mushroom (Schlimmer, 1987) | 8124 | 22 | 2 | https://archive.ics.uci.edu/ml/datasets/Mushroom |
| 6 | SPECT Heart (Kurgan ve diğ., 2001) | 267 | 22 | 2 | https://archive.ics.uci.edu/ml/datasets/SPECT+Heart |
| 7 | Chess (King-Rook vs. King-Pawn) (Shapiro, 1984) | 3196 | 36 | 2 | https://archive.ics.uci.edu/ml/datasets/Chess+%28King-Rook+vs.+King-Pawn%29 |
| 8 | Splice-junction Gene Sequences (Noordewier ve diğ., 1990) | 3190 | 61 | 3 | https://archive.ics.uci.edu/ml/datasets/Molecular+Biology+%28Splice-junction+Gene+Sequences%29 |
| 9 | Audiology (Standardized) (Bareiss ve diğ., 1988) | 226 | 69 | 24 | https://archive.ics.uci.edu/ml/datasets/Audiology+%28Standardized%29 |
| 10 | Soybean (Michalski ve Chilausky, 1980) | 307 | 35 | 19 | https://archive.ics.uci.edu/ml/datasets/Soybean+(Large) |

## 3.3 KULLANILAN ARAÇLAR

Uygulamada Python 3.8 yazılım dili, colab standart versiyonu (colab pro veya colab pro+ değil) ve çok amaçlı optimizasyon için jMetalPy kütüphanesi kullanılmıştır (Benítez-Hidalgo ve diğ., 2019).

İlk olarak, UCI veri deposundan alınan ve Tablo 3.1'de listelenen veri kümeleri dikkatlice incelenmiştir. Veri ön işleme aşamasında, yalnızca eksik değerleri içeren Breast cancer, Congressional Voting Records, Primary Tumor, Mushroom, Audiology, and Soybean veri

kümelerinde imputasyon işlemi yapılmıştır. Veri kümeleri tamamen kategorik değişkenlerden oluştuğu için, kategorik değişkenler için en sade imputasyon yaklaşımı olan mod değerinin atanması tercih edilmiştir.

ÇOKOTUÖS yaklaşımı için Bölüm 3.1'de ifade edilen adımlar sırayla uygulanmış ve aşağıdaki sonuçlar elde edilmiştir:

1- Kesikleştirme: Bu tez çalışması kapsamında, kesikleştirme performansının öznitelik seçim performansını etkileme olasılığı olduğundan, tez çalışmasına dahil edilen tüm veri kümeleri tamamen kategorik özniteliklerden oluşan kümelerden seçilmiştir.

2- Alt Küme Uzayının Tanımlanması: Bu aşamada Breast veri seti için kapsamlı arama ile 510 aday öznitelik alt kümesi ve diğer veri kümeleri için rastgele arama ile belirlenen 500 aday öznitelik alt kümesi üzerinde değerlendirme yapılır.

3- Meritlerin Hesaplanması: Bir önceki aşamada belirlenen her aday öznitelik alt kümesi için CFS ve CONS merit değerleri hesaplanır ve öznitelik adlarıyla birlikte bir tabloda saklanır.

4- Optimizasyon: Çok amaçlı optimizasyon yaklaşımları (Pareto ve NSGA II) uygulanarak hesaplanan merit değerlerinin maksimize edilmesi amacıyla aday öznitelik alt kümeleri belirlenir. ÇOKOTUÖS-NSGA II ve ÇOKOTUÖS-Pareto tarafından belirlenen adaylar yeni bir tabloda saklanır.

5- Sarmalayıcı: Bir önceki aşamada belirlenen her öznitelik alt kümesi adayı için CatBoost algoritması ile sınıflandırma yapılır ve tahmini doğruluk değeri tabloda saklanır.

6- Seçilmiş Öznitelik Alt kümesi: ÇOKOTUÖS-NSGA II ve ÇOKOTUÖS-Pareto yaklaşımları kapsamında değerlendirilen aday öznitelik alt kümeleri arasından predictive doğruluk değerine göre en yüksek puanı üreten aday seçilir. Birden fazla adayın en yüksek doğruluk değerini alması durumunda öznitelik sayısı en az olan aday seçilmiştir. Hem doğruluk değeri hem de öznitelik sayısının eşit olduğu durumlarda ise adaylardan biri rastgele seçilmiştir.

Bu uygulamada jMetalPy kütüphanesindeki NSGA II fonksiyonu için kullanılan parametreler aşağıda yer almaktadır:

(i) `population_size=100,`

(ii) `offspring_population_size=1,`

(iii) `mutation=BitFlipMutation(probability= 1.0),`

(iv) `crossover =SPXCrossover(probability = 1.0),`

(v) `termination_criterion=StoppingByEvaluations(max_evaluations=2500),`

(vi) `dominance_comparator=DominanceComparator().`

**Tablo 3.2:** Tez Çalışmasında kullanılan sınıflandırıcılar.

| | Sınıflandırıcı |
|---|---|
| 1 | Catboost |
| 2 | DecisionTreeClassifier |
| 3 | ExtraTreesClassifier |
| 4 | GaussianNB |
| 5 | KNeighborsClassifier |
| 6 | LGBMClassifier |
| 7 | LogisticRegression |
| 8 | PassiveAggressiveClassifier |
| 9 | RandomForestClassifier |
| 10 | SGDClassifier |
| 11 | SVC |
| 12 | XGBClassifier |

CatBoost, sarmalayıcı sınıflandırıcı olarak kullanılır. Tablo 3.1'deki her bir veri kümesi için CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGA II yaklaşımları kullanılarak dört öznitelik seçim uygulaması yapılmış ve elde edilen öznitelik alt kümeleri için 12 farklı sınıflandırıcı yardımıyla tahminler yapılmıştır. Kullanılan sınıflandırma algoritmaları Tablo 3.2'de gösterilmiştir.

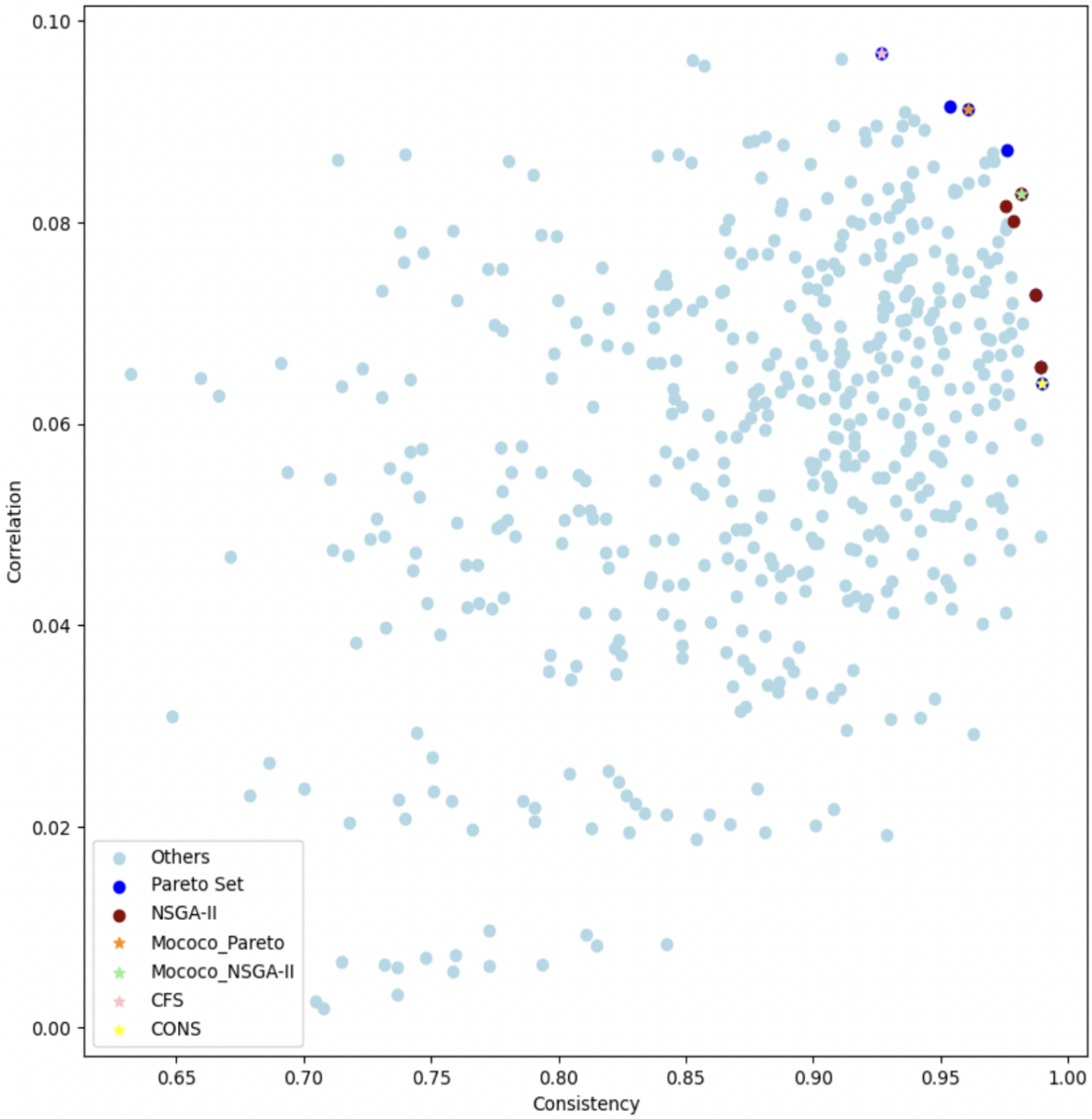
Sınıflandırma doğruluğu *(accuracy)* hesaplaması için %70 öğrenme *(train),* %30 test olacak şekilde *hold-out* yaklaşımı kullanılmıştır. Kullanılan her bir sınıflandırma algoritmasının aynı test ve öğrenme veri kümelerini kullanması sağlanmıştır.

# 4. BULGULAR

Bu bölümde, yapılan tez çalışmasından elde edilen sonuçlar merit değerlerine göre elde edilen dağılımlar, ortalama performanslara göre karşılaştırmalar, en yüksek performanslara göre karşılaştırmalar, en yüksek performanslarına göre sıralamalar ve F1 skor standart sapmalarına göre karşılaştırmalar üzerinden anlatılmıştır.

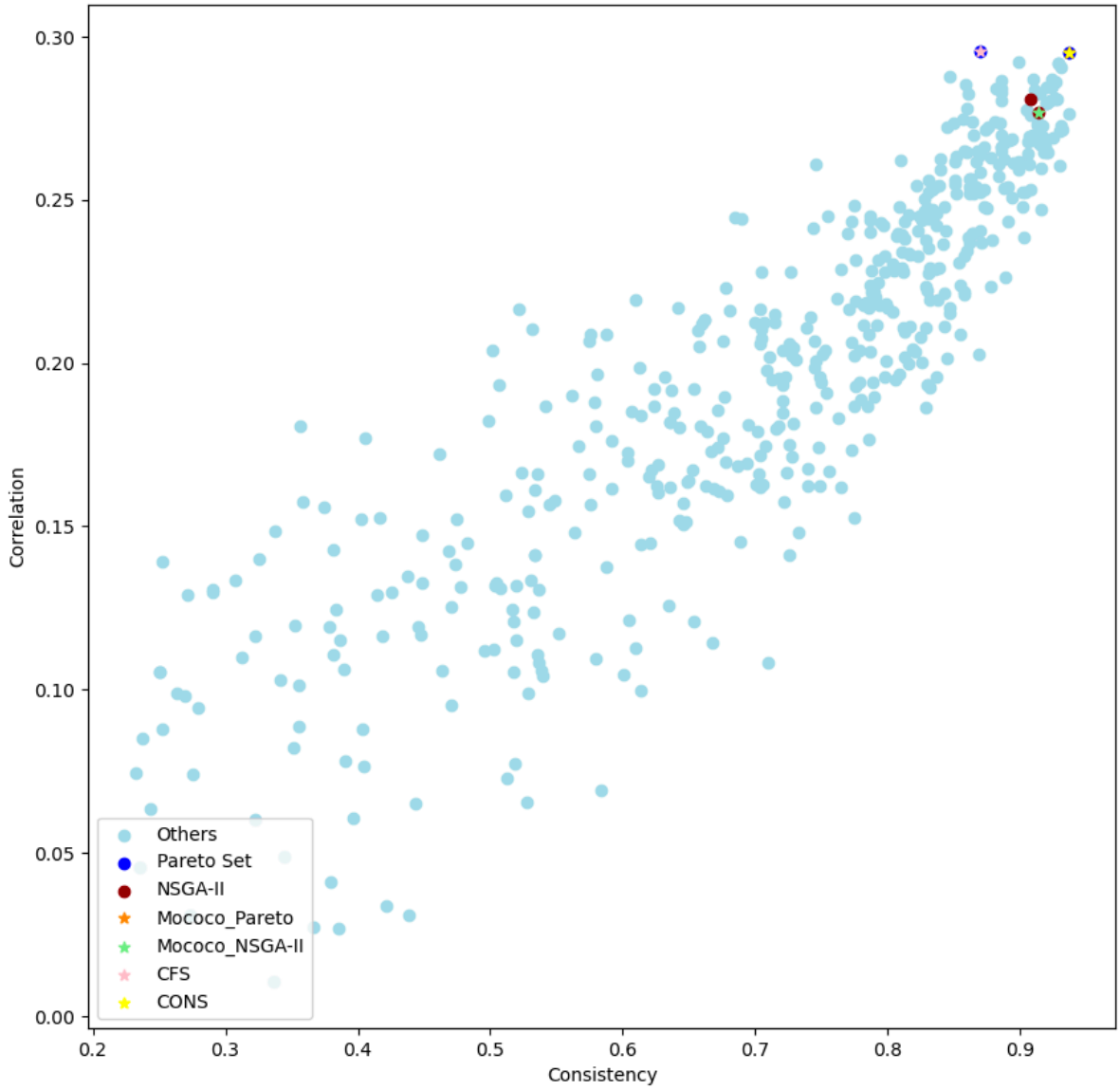
## 4.1 MERİT DEĞERLERİNİN DAĞILIM GRAFİKLERİ

Kartezyen düzlemde her veri kümesinden seçilen aday öznitelik alt kümeleri için hesaplanan CFS ve CONS başarı değerlerinin gösterimi Şekil 4.1- 4.10'da verilmiştir. Ayrıca bu şekillerde ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGA II tarafından belirlenen adaylar koyu renklerle gösterilmiştir. CFS, CONS, ÇOKOTUÖS-NSGA II ve ÇOKOTUÖS-Pareto tarafından seçilen öznitelik alt kümeleri yıldızlarla gösterilmiştir.



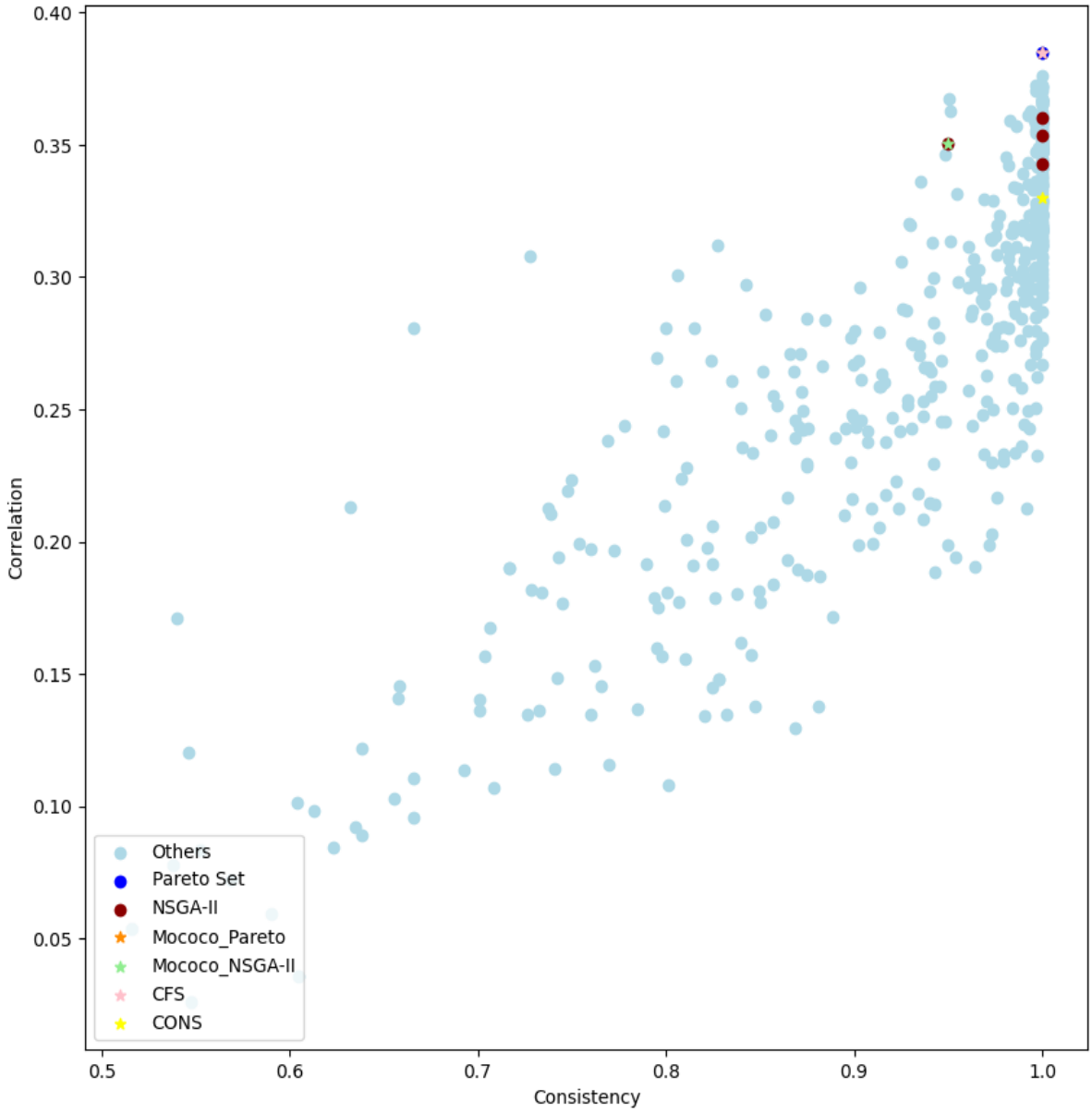

**Şekil 4.1:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi – Breast.

Şekil 4.1'de Breast veri kümesi için, değerlendirilen öznitelik alt kümesi adaylarına ait CFS ve CONS merit değerlerinin oluşturduğu kartezyen düzlem gösterilmiştir. Ayrıca bu düzlemde yer alan adaylardan Pareto ve NSGA-II yaklaşımları ile filtrelenen adaylar koyu renkli yuvarlaklarla gösterilmiştir. CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümeleri ise kartezyen düzlem üzerinde yıldızlarla gösterilmiştir.



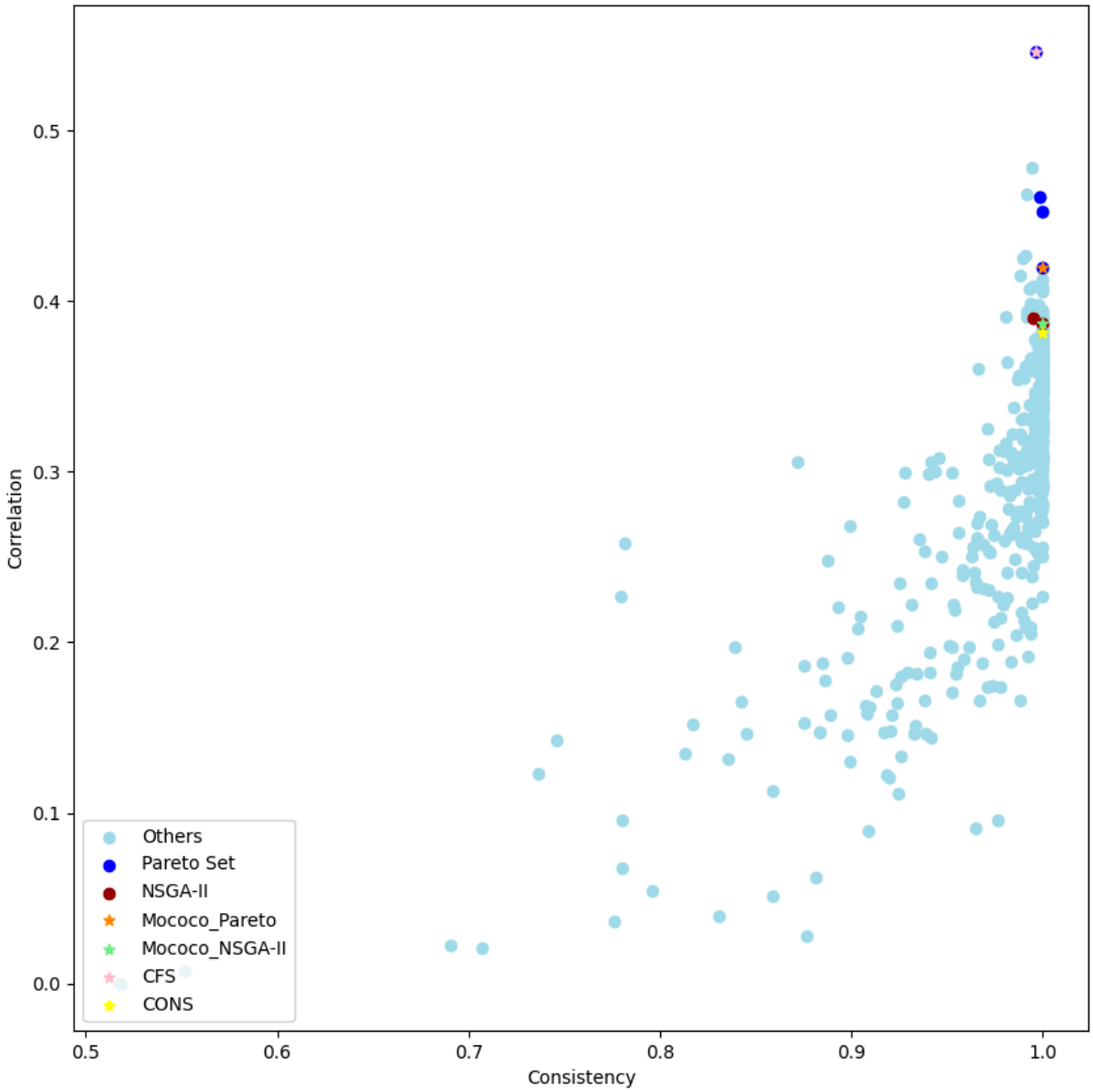

**Şekil 4.2:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi - Primary Tumor.

Şekil 4.2'de Primary Tumor veri kümesi için, değerlendirilen öznitelik alt kümesi adaylarına ait CFS ve CONS merit değerlerinin oluşturduğu kartezyen düzlem gösterilmiştir. Ayrıca bu düzlemde yer alan adaylardan Pareto ve NSGA-II yaklaşımları ile filtrelenen adaylar koyu renkli yuvarlaklarla gösterilmiştir. CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümeleri ise kartezyen düzlem üzerinde yıldızlarla gösterilmiştir.



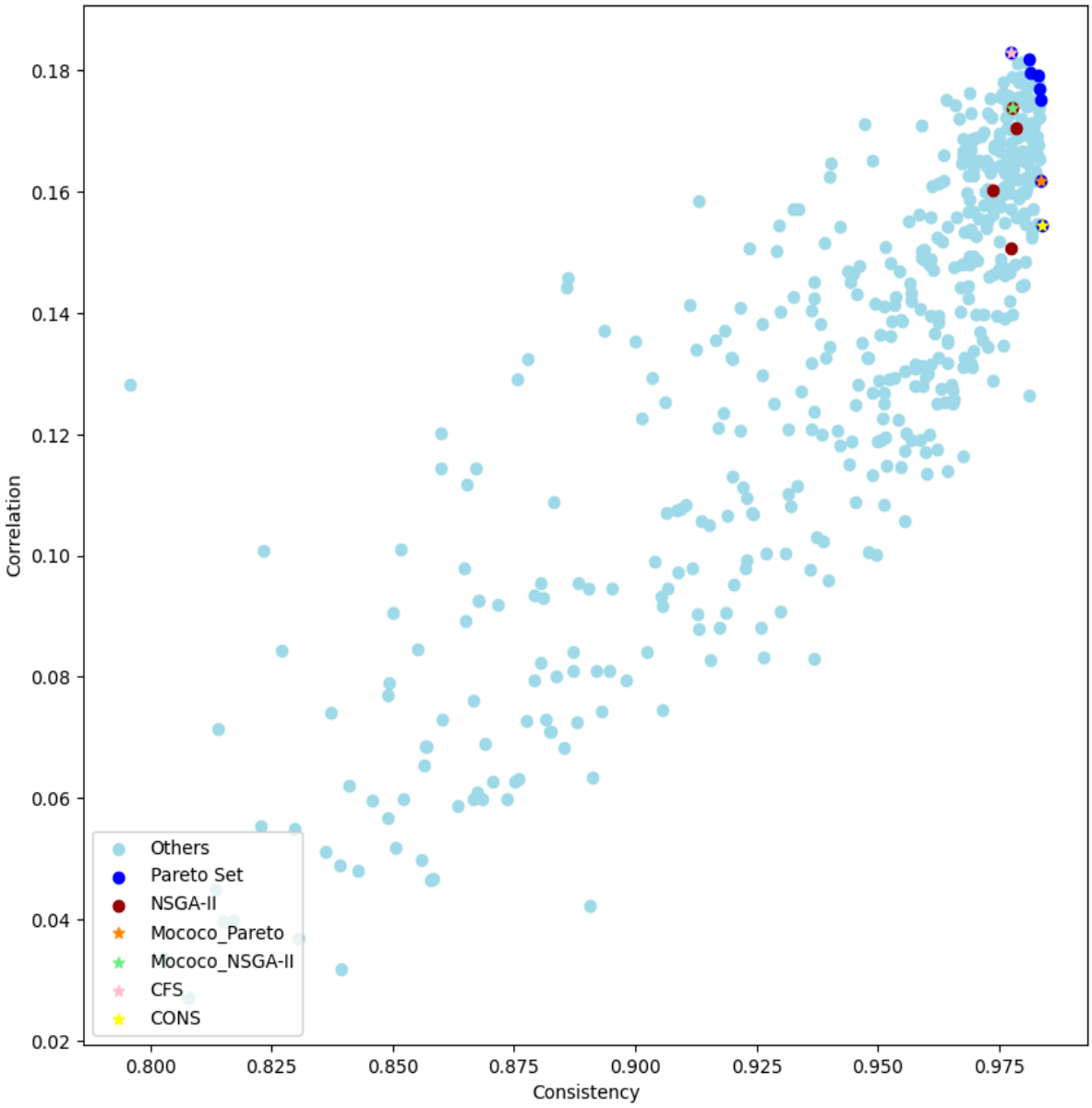

**Şekil 4.3:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi – Lymphography.

Şekil 4.3'te Lymphography veri kümesi için, değerlendirilen öznitelik alt kümesi adaylarına ait CFS ve CONS merit değerlerinin oluşturduğu kartezyen düzlem gösterilmiştir. Ayrıca bu düzlemde yer alan adaylardan Pareto ve NSGA-II yaklaşımları ile filtrelenen adaylar koyu renkli yuvarlaklarla gösterilmiştir. CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümeleri ise kartezyen düzlem üzerinde yıldızlarla gösterilmiştir.



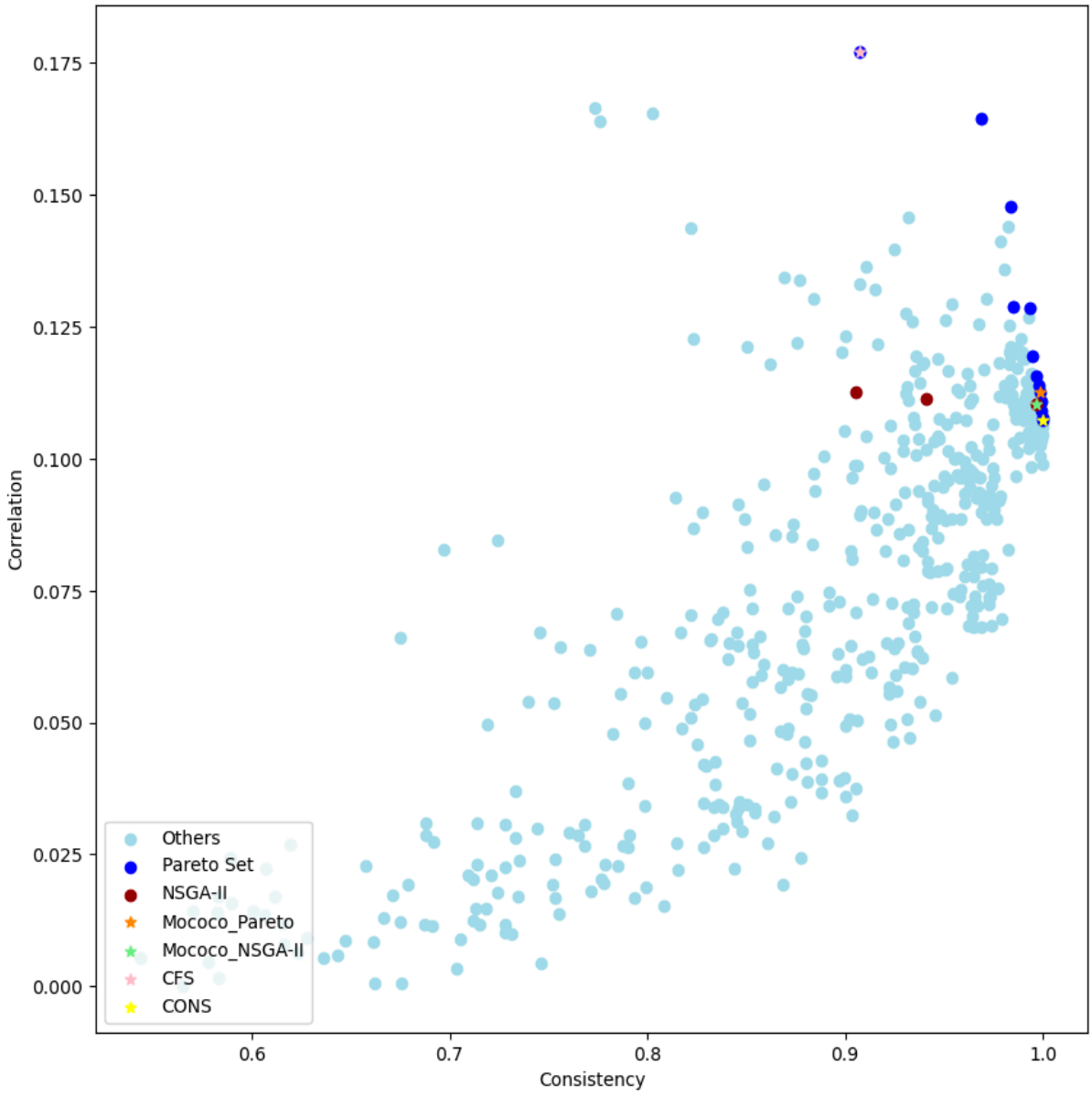

**Şekil 4.4:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi – Mushroom.

Şekil 4.4'te Mushroom veri kümesi için, değerlendirilen öznitelik alt kümesi adaylarına ait CFS ve CONS merit değerlerinin oluşturduğu kartezyen düzlem gösterilmiştir. Ayrıca bu düzlemde yer alan adaylardan Pareto ve NSGA-II yaklaşımları ile filtrelenen adaylar koyu renkli yuvarlaklarla gösterilmiştir. CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümeleri ise kartezyen düzlem üzerinde yıldızlarla gösterilmiştir.



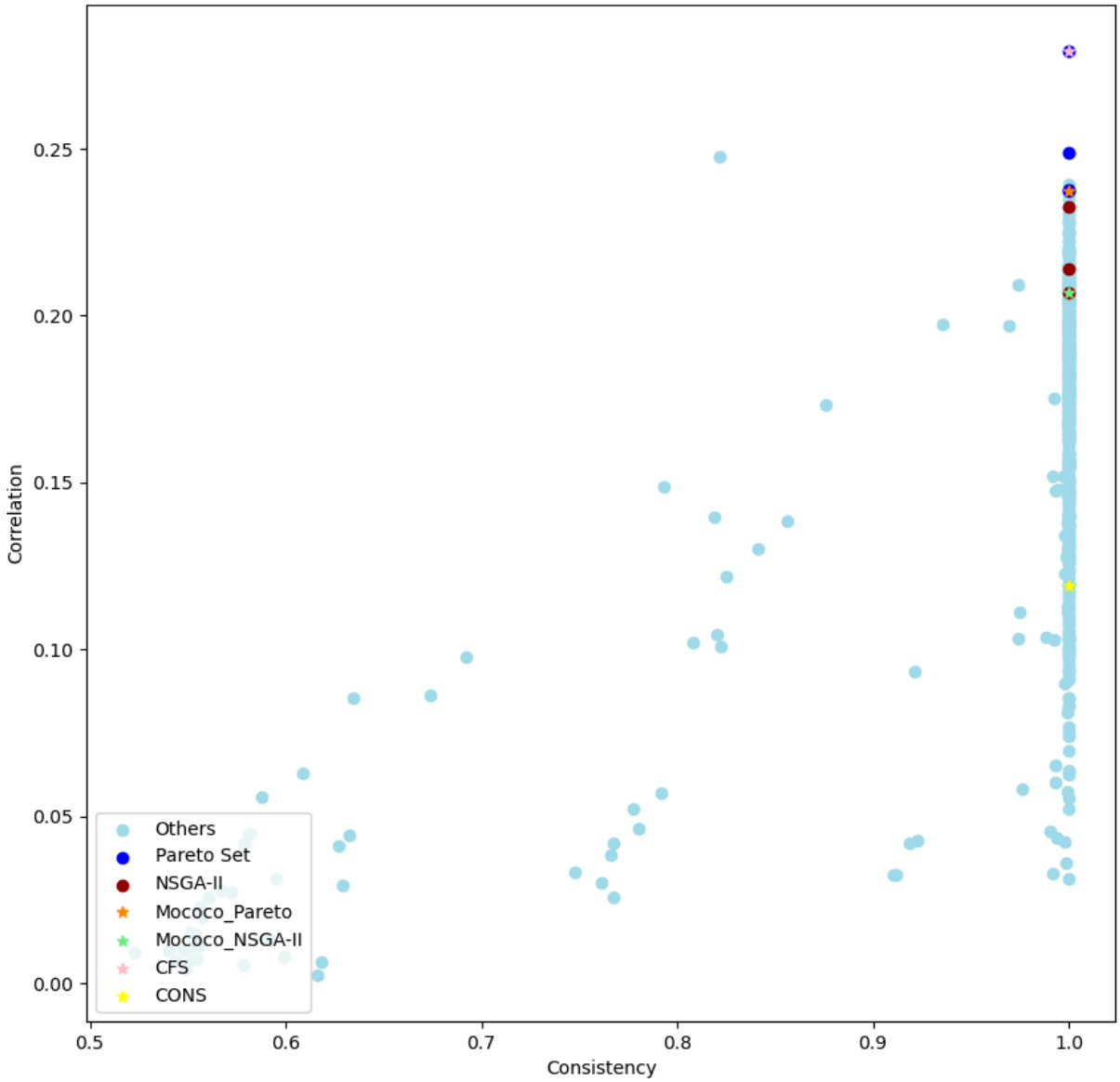

**Şekil 4.5:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi – Spect Heart.

Şekil 4.5'te Spect Heart veri kümesi için, değerlendirilen öznitelik alt kümesi adaylarına ait CFS ve CONS merit değerlerinin oluşturduğu kartezyen düzlem gösterilmiştir. Ayrıca bu düzlemde yer alan adaylardan Pareto ve NSGA-II yaklaşımları ile filtrelenen adaylar koyu renkli yuvarlaklarla gösterilmiştir. CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümeleri ise kartezyen düzlem üzerinde yıldızlarla gösterilmiştir.



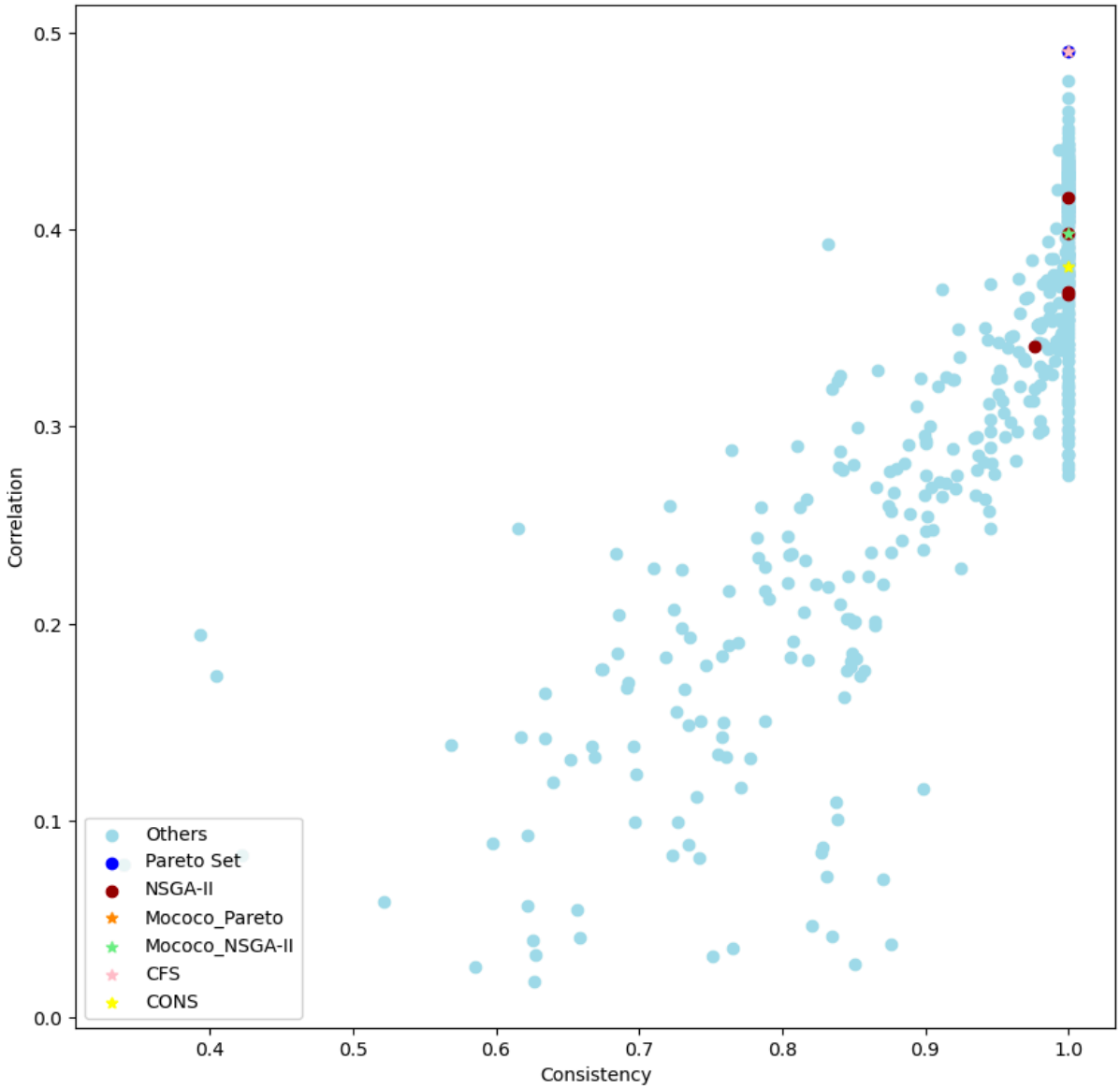

**Şekil 4.6:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi – Chess.

Şekil 4.6'da Chess veri kümesi için, değerlendirilen öznitelik alt kümesi adaylarına ait CFS ve CONS merit değerlerinin oluşturduğu kartezyen düzlem gösterilmiştir. Ayrıca bu düzlemde yer alan adaylardan Pareto ve NSGA-II yaklaşımları ile filtrelenen adaylar koyu renkli yuvarlaklarla gösterilmiştir. CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümeleri ise kartezyen düzlem üzerinde yıldızlarla gösterilmiştir.



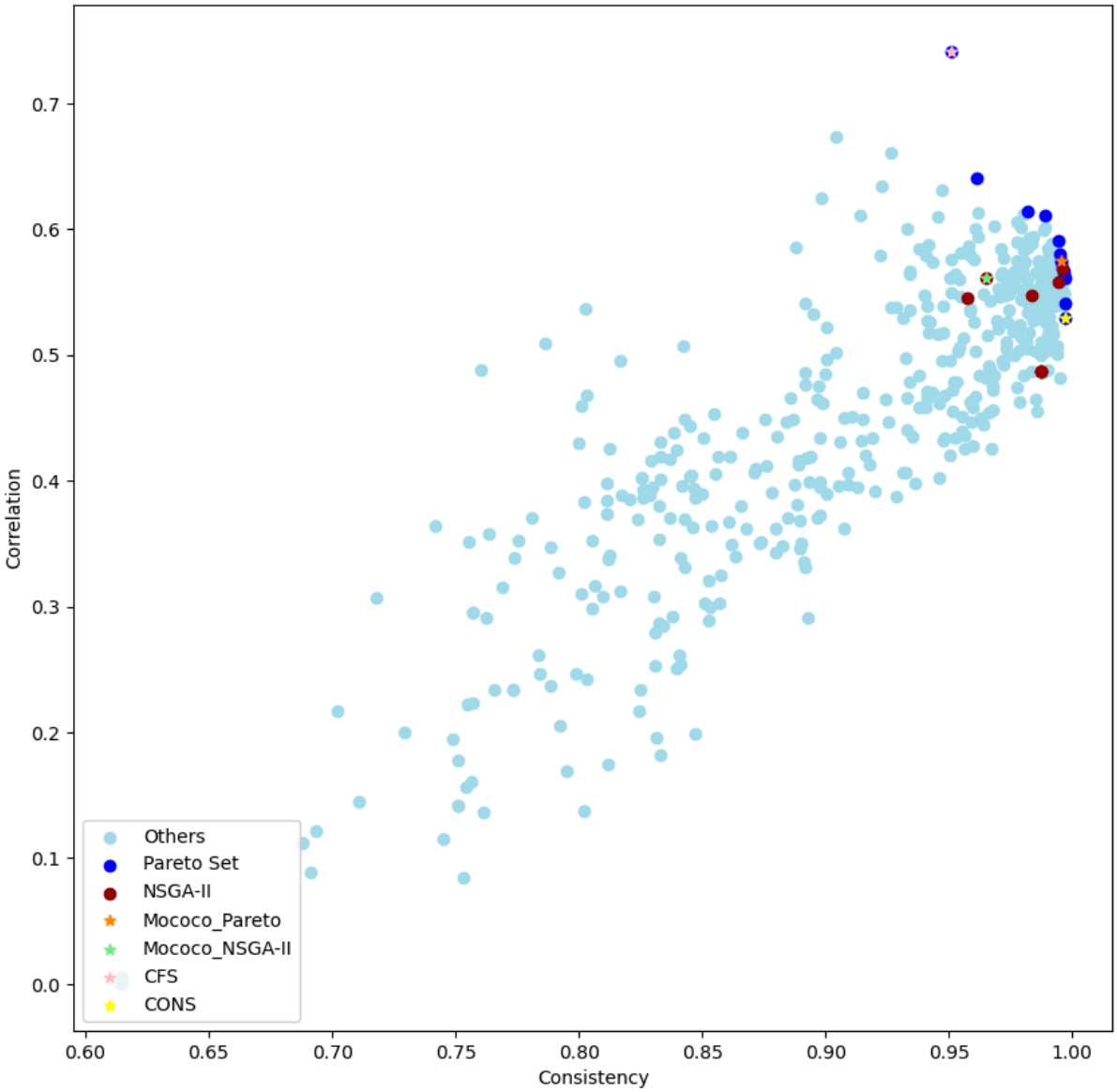

**Şekil 4.7:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi – Splice.

Şekil 4.7'de Splice veri kümesi için, değerlendirilen öznitelik alt kümesi adaylarına ait CFS ve CONS merit değerlerinin oluşturduğu kartezyen düzlem gösterilmiştir. Ayrıca bu düzlemde yer alan adaylardan Pareto ve NSGA-II yaklaşımları ile filtrelenen adaylar koyu renkli yuvarlaklarla gösterilmiştir. CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümeleri ise kartezyen düzlem üzerinde yıldızlarla gösterilmiştir.



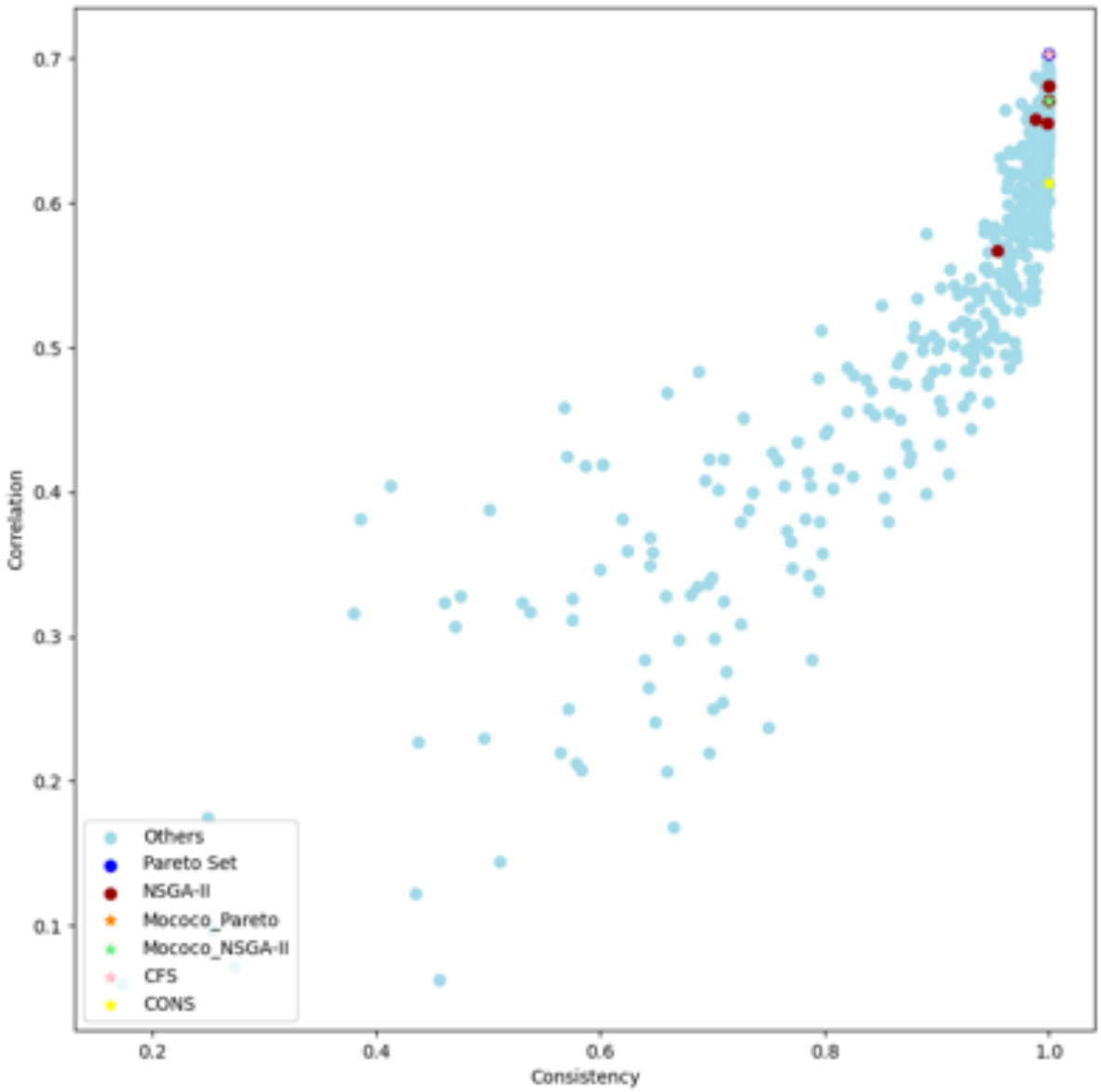

**Şekil 4.8:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi – Audiology.

Şekil 4.8'de Audiology veri kümesi için, değerlendirilen öznitelik alt kümesi adaylarına ait CFS ve CONS merit değerlerinin oluşturduğu kartezyen düzlem gösterilmiştir. Ayrıca bu düzlemde yer alan adaylardan Pareto ve NSGA-II yaklaşımları ile filtrelenen adaylar koyu renkli yuvarlaklarla gösterilmiştir. CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümeleri ise kartezyen düzlem üzerinde yıldızlarla gösterilmiştir.




**Şekil 4.9:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi - Voting Records.

Şekil 4.9'da Voting Records veri kümesi için, değerlendirilen öznitelik alt kümesi adaylarına ait CFS ve CONS merit değerlerinin oluşturduğu kartezyen düzlem gösterilmiştir. Ayrıca bu düzlemde yer alan adaylardan Pareto ve NSGA-II yaklaşımları ile filtrelenen adaylar koyu renkli yuvarlaklarla gösterilmiştir. CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümeleri ise kartezyen düzlem üzerinde yıldızlarla gösterilmiştir.




**Şekil 4.10:** Değerlendirilen öznitelik alt kümelerinin korelasyon ve tutarlılık merit değerleri kartezyen düzlemi – Soybean.

Şekil 4.10'da Soybean veri kümesi için, değerlendirilen öznitelik alt kümesi adaylarına ait CFS ve CONS merit değerlerinin oluşturduğu kartezyen düzlem gösterilmiştir. Ayrıca bu düzlemde yer alan adaylardan Pareto ve NSGA-II yaklaşımları ile filtrelenen adaylar koyu renkli yuvarlaklarla gösterilmiştir. CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümeleri ise kartezyen düzlem üzerinde yıldızlarla gösterilmiştir.

## 4.2 ORTALAMALARA GÖRE KARŞILAŞTIRMA

EK-3 ile detaylı sınıflandırma sonuçları paylaşılmıştır ve elde edilen sonuçların ortalamalarına yönelik analizler bu bölümde incelenmiştir. Tablo 3.1'de verilen on veri seti için ÇOKOTUÖS-

NSGA II, ÇOKOTUÖS-Pareto, CFS ve CONS tarafından seçilen öznitelik alt kümelerinin, Tablo 3.2'de belirtilen algoritmalar kullanılarak yapılan sınıflandırma çalışmaları sonucunda elde edilen sınıflandırma doğruluğu ve F1 skor değerlerinin ortalama ve standart sapmaları Tablo 4.1- 4.10'da verilmiştir.

Tablo 4.1'de Audiology veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen sınıflandırma doğruluğu ve F1 skor sonuçlarının ortalaması gösterilmiştir. Ayrıca öznitelik alt kümesinin eleman sayısı ve yaklaşımların sonuç vermesi için gerek duyduğu sarmalayıcı yaklaşım sayısı da yine Tablo 4.1 üzerinde gösterilmiştir.

**Tablo 4.1:** Ortalama karşılaştırması - Audiology veri kümesi.

| Seçim Yöntemi | Sınıflandırma Doğruluğu | F1 Skor | Öznitelik Adedi | Sarmalayıcı Adedi |
|---|---|---|---|---|
| ÇOKOTUÖS-NSGA II | 0.6658 ± 0.1243 | 0.6237 ± 0.1443 | 60 | 4 |
| ÇOKOTUÖS-Pareto | 0.639 ± 0.1318 | 0.5871 ± 0.1688 | 22 | 0 |
| CFS | 0.639 ± 0.1318 | 0.5871 ± 0.1688 | 22 | 0 |
| CONS | 0.6337 ± 0.1565 | 0.5807 ± 0.1945 | 3 | 0 |

Tablo 4.2'de Breast veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen sınıflandırma doğruluğu ve F1 skor sonuçlarının ortalaması gösterilmiştir. Ayrıca öznitelik alt kümesinin eleman sayısı ve yaklaşımların sonuç vermesi için gerek duyduğu sarmalayıcı yaklaşım sayısı da yine Tablo 4.2 üzerinde gösterilmiştir.

**Tablo 4.2:** Ortalama karşılaştırması - Breast veri kümesi.

| Seçim Yöntemi | Sınıflandırma Doğruluğu | F1 Skor | Öznitelik Adedi | Sarmalayıcı Adedi |
|---|---|---|---|---|
| ÇOKOTUÖS-Pareto | 0.6765 ± 0.0557 | 0.6606 ± 0.0511 | 6 | 7 |
| CFS | 0.6734 ± 0.0521 | 0.6553 ± 0.0515 | 5 | 0 |
| CONS | 0.6712 ± 0.0942 | 0.6493 ± 0.103 | 8 | 0 |
| ÇOKOTUÖS-NSGA II | 0.6691 ± 0.099 | 0.6465 ± 0.1005 | 8 | 4 |

Tablo 4.3'te Chess veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen sınıflandırma doğruluğu ve F1 skor sonuçlarının ortalaması gösterilmiştir. Ayrıca öznitelik alt kümesinin eleman sayısı ve yaklaşımların sonuç vermesi için gerek duyduğu sarmalayıcı yaklaşım sayısı da yine Tablo 4.3 üzerinde gösterilmiştir.

**Tablo 4.3:** Ortalama karşılaştırması - Chess veri kümesi.

| Seçim Yöntemi | Sınıflandırma Doğruluğu | F1 Skor | Öznitelik Adedi | Sarmalayıcı Adedi |
|---|---|---|---|---|
| CFS | 0.9411 ± 0.1044 | 0.936 ± 0.1213 | 4 | 0 |
| ÇOKOTUÖS-NSGA II | 0.9407 ± 0.1112 | 0.9341 ± 0.1325 | 30 | 2 |
| ÇOKOTUÖS-Pareto | 0.9404 ± 0.1132 | 0.9335 ± 0.1357 | 32 | 12 |
| CONS | 0.9404 ± 0.1103 | 0.934 ± 0.1311 | 34 | 0 |

Tablo 4.4'te Lymphography veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen sınıflandırma doğruluğu ve F1 skor sonuçlarının ortalaması gösterilmiştir. Ayrıca öznitelik alt kümesinin eleman sayısı ve yaklaşımların sonuç vermesi için gerek duyduğu sarmalayıcı yaklaşım sayısı da yine Tablo 4.4 üzerinde gösterilmiştir.

**Tablo 4.4:** Ortalama karşılaştırması - Lymphography veri kümesi.

| Seçim Yöntemi | Sınıflandırma Doğruluğu | F1 Skor | Öznitelik Adedi | Sarmalayıcı Adedi |
|---|---|---|---|---|
| CONS | 0.8182 ± 0.0618 | 0.8123 ± 0.0637 | 10 | 0 |
| ÇOKOTUÖS-NSGA II | 0.814 ± 0.0404 | 0.8041 ± 0.0416 | 6 | 3 |
| ÇOKOTUÖS-Pareto | 0.8099 ± 0.0559 | 0.8055 ± 0.056 | 11 | 0 |
| CFS | 0.8099 ± 0.0559 | 0.8055 ± 0.056 | 11 | 0 |

Tablo 4.5'te Mushroom veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen sınıflandırma doğruluğu ve F1 skor sonuçlarının ortalaması gösterilmiştir. Ayrıca

öznitelik alt kümesinin eleman sayısı ve yaklaşımların sonuç vermesi için gerek duyduğu sarmalayıcı yaklaşım sayısı da yine Tablo 4.5 üzerinde gösterilmiştir.

**Tablo 4.5:** Ortalama karşılaştırması - Mushroom veri kümesi.

| Seçim Yöntemi | Sınıflandırma Doğruluğu | F1 Skor | Öznitelik Adedi | Sarmalayıcı Adedi |
|---|---|---|---|---|
| ÇOKOTUÖS-Pareto | 0.9998 ± 0.0007 | 0.9998 ± 0.0007 | 8 | 3 |
| ÇOKOTUÖS-NSGA II | 0.9993 ± 0.0021 | 0.9993 ± 0.0021 | 16 | 2 |
| CFS | 0.9986 ± 0.0046 | 0.9986 ± 0.0046 | 1 | 0 |
| CONS | 0.9798 ± 0.0664 | 0.9791 ± 0.0688 | 7 | 0 |

Tablo 4.6'da Primary Tumor veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen sınıflandırma doğruluğu ve F1 skor sonuçlarının ortalaması gösterilmiştir. Ayrıca öznitelik alt kümesinin eleman sayısı ve yaklaşımların sonuç vermesi için gerek duyduğu sarmalayıcı yaklaşım sayısı da yine Tablo 4.6 üzerinde gösterilmiştir.

**Tablo 4.6:** Ortalama karşılaştırması - Primary Tumor veri kümesi.

| Seçim Yöntemi | Sınıflandırma Doğruluğu | F1 Skor | Öznitelik Adedi | Sarmalayıcı Adedi |
|---|---|---|---|---|
| CFS | 0.328 ± 0.0398 | 0.3076 ± 0.0444 | 13 | 0 |
| ÇOKOTUÖS-NSGA II | 0.3262 ± 0.0517 | 0.3064 ± 0.0565 | 15 | 2 |
| CONS | 0.3253 ± 0.0507 | 0.3013 ± 0.0548 | 15 | 0 |
| ÇOKOTUÖS-Pareto | 0.3253 ± 0.0507 | 0.3013 ± 0.0548 | 15 | 2 |

Tablo 4.7'de Soybean veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen sınıflandırma doğruluğu ve F1 skor sonuçlarının ortalaması gösterilmiştir. Ayrıca öznitelik alt kümesinin eleman sayısı ve yaklaşımların sonuç vermesi için gerek duyduğu sarmalayıcı yaklaşım sayısı da yine Tablo 4.7 üzerinde gösterilmiştir.

**Tablo 4.7:** Ortalama karşılaştırması - Soybean veri kümesi.

| Seçim Yöntemi | Sınıflandırma Doğruluğu | F1 Skor | Öznitelik Adedi | Sarmalayıcı Adedi |
|---|---|---|---|---|
| CONS | 0.8844 ± 0.0334 | 0.881 ± 0.0364 | 19 | 0 |
| ÇOKOTUÖS-NSGA II | 0.8834 ± 0.03 | 0.8763 ± 0.0335 | 33 | 6 |
| ÇOKOTUÖS-Pareto | 0.8814 ± 0.0324 | 0.8729 ± 0.0361 | 31 | 0 |
| CFS | 0.8814 ± 0.0324 | 0.8729 ± 0.0361 | 31 | 0 |

Tablo 4.8'de Spect Heart veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen sınıflandırma doğruluğu ve F1 skor sonuçlarının ortalaması gösterilmiştir. Ayrıca öznitelik alt kümesinin eleman sayısı ve yaklaşımların sonuç vermesi için gerek duyduğu sarmalayıcı yaklaşım sayısı da yine Tablo 4.8 üzerinde gösterilmiştir.

**Tablo 4.8:** Ortalama karşılaştırması - Spect Heart veri kümesi.

| Seçim Yöntemi | Sınıflandırma Doğruluğu | F1 Skor | Öznitelik Adedi | Sarmalayıcı Adedi |
|---|---|---|---|---|
| CONS | 0.7193 ± 0.0513 | 0.7032 ± 0.0611 | 19 | 0 |
| ÇOKOTUÖS-NSGA II | 0.717 ± 0.0482 | 0.7158 ± 0.0404 | 17 | 4 |
| ÇOKOTUÖS-Pareto | 0.7148 ± 0.0533 | 0.6986 ± 0.0614 | 19 | 8 |
| CFS | 0.7148 ± 0.0515 | 0.7154 ± 0.0427 | 13 | 0 |

Tablo 4.9'da Splice veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen sınıflandırma doğruluğu ve F1 skor sonuçlarının ortalaması gösterilmiştir. Ayrıca öznitelik alt kümesinin eleman sayısı ve yaklaşımların sonuç vermesi için gerek duyduğu sarmalayıcı yaklaşım sayısı da yine Tablo 4.9 üzerinde gösterilmiştir.

**Tablo 4.9:** Ortalama karşılaştırması - Splice veri kümesi.

| Seçim Yöntemi | Sınıflandırma Doğruluğu | F1 Skor | Öznitelik Adedi | Sarmalayıcı Adedi |
|---|---|---|---|---|
| CFS | 0.9305 ± 0.0944 | 0.9228 ± 0.114 | 18 | 0 |

| | | | | |
|---|---|---|---|---|
| ÇOKOTUÖS-Pareto | 0.8076 ± 0.2099 | 0.7891 ± 0.2436 | 33 | 4 |
| ÇOKOTUÖS-NSGA II | 0.793 ± 0.2175 | 0.7753 ± 0.2493 | 52 | 3 |
| CONS | 0.7085 ± 0.2393 | 0.6701 ± 0.2885 | 5 | 0 |

Tablo 4.10'da Voting Records veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen sınıflandırma doğruluğu ve F1 skor sonuçlarının ortalaması gösterilmiştir. Ayrıca öznitelik alt kümesinin eleman sayısı ve yaklaşımların sonuç vermesi için gerek duyduğu sarmalayıcı yaklaşım sayısı da yine Tablo 4.10 üzerinde gösterilmiştir.

**Tablo 4.10:** Ortalama karşılaştırması - Voting Records veri kümesi.

| Seçim Yöntemi | Sınıflandırma Doğruluğu | F1 Skor | Öznitelik Adedi | Sarmalayıcı Adedi |
|---|---|---|---|---|
| CFS | 0.9755 ± 0.0128 | 0.9755 ± 0.0128 | 1 | 0 |
| ÇOKOTUÖS-NSGA II | 0.972 ± 0.0183 | 0.972 ± 0.0184 | 7 | 8 |
| CONS | 0.972 ± 0.0169 | 0.972 ± 0.017 | 13 | 0 |
| ÇOKOTUÖS-Pareto | 0.9685 ± 0.0193 | 0.9686 ± 0.0193 | 13 | 14 |

## 4.3 EN YÜKSEK PERFORMANSA GÖRE KARŞILAŞTIRMA

Tablo 3.1'deki 10 veri seti için Tablo 3.2'de belirtilen sınıflandırıcılar kullanılarak ÇOKOTUÖS-NSGAII, ÇOKOTUÖS-Pareto, CFS ve CONS tarafından seçilen öznitelik alt kümelerinin sınıflandırma çalışmaları sonucunda elde edilen en yüksek doğruluk değerleri Tablo 4.11- 4.20'de verilmiştir.

Tablo 4.11 ile Audiology veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen en yüksek sınıflandırma doğrulukları ve sınıflandırma doğruluğuna göre büyükten küçüğe doğru kaçıncı sırada olduğu gösterilmiştir.

Tablo 4.12 ile Breast veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak

elde edilen en yüksek sınıflandırma doğrulukları ve sınıflandırma doğruluğuna göre büyükten küçüğe doğru kaçıncı sırada olduğu gösterilmiştir.

**Tablo 4.11:** En yüksek doğruluk karşılaştırması - Audiology veri kümesi.

| Veri kümesi | Seçim Yöntemi | Sınıflandırma Doğruluğu | Sıralama |
|---|---|---|---|
| Audiology | CONS | 83,82 | 1 |
| Audiology | ÇOKOTUÖS-NSGAII | 82,35 | 2 |
| Audiology | ÇOKOTUÖS-Pareto | 79,41 | 4 |
| Audiology | CFS | 79,41 | 4 |

**Tablo 4.12:** En yüksek doğruluk karşılaştırması - Breast veri kümesi.

| Veri kümesi | Seçim Yöntemi | Sınıflandırma Doğruluğu | Sıralama |
|---|---|---|---|
| Breast | ÇOKOTUÖS-NSGAII | 74,42 | 1 |
| Breast | ÇOKOTUÖS-Pareto | 73,26 | 2 |
| Breast | CONS | 73,26 | 2 |
| Breast | CFS | 72,09 | 4 |

Tablo 4.13 ile Chess veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGA II ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen en yüksek sınıflandırma doğrulukları ve sınıflandırma doğruluğuna göre büyükten küçüğe doğru kaçıncı sırada olduğu gösterilmiştir.

**Tablo 4.13**: En yüksek doğruluk karşılaştırması - Chess veri kümesi.

| Veri kümesi | Seçim Yöntemi | Sınıflandırma Doğruluğu | Sıralama |
|---|---|---|---|
| Chess | ÇOKOTUÖS-Pareto | 99,27 | 2 |
| Chess | CONS | 99,27 | 2 |
| Chess | ÇOKOTUÖS-NSGAII | 99,06 | 4 |
| Chess | CFS | 99,06 | 4 |

Tablo 4.14 ile Lymphography veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen en yüksek sınıflandırma doğrulukları ve sınıflandırma doğruluğuna göre büyükten küçüğe doğru kaçıncı sırada olduğu gösterilmiştir.

**Tablo 4.14:** En yüksek doğruluk karşılaştırması - Lymphography veri kümesi.

| Veri kümesi | Seçim Yöntemi | Sınıflandırma Doğruluğu | Sıralama |
|---|---|---|---|
| Lymphography | CONS | 93,18 | 1 |
| Lymphography | ÇOKOTUÖS-Pareto | 88,64 | 4 |
| Lymphography | ÇOKOTUÖS-NSGAII | 88,64 | 4 |
| Lymphography | CFS | 88,64 | 4 |

Tablo 4.15 ile Mushroom veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen en yüksek sınıflandırma doğrulukları ve sınıflandırma doğruluğuna göre büyükten küçüğe doğru kaçıncı sırada olduğu gösterilmiştir.

**Tablo 4.15:** En yüksek doğruluk karşılaştırması - Mushroom veri kümesi.

| Veri kümesi | Seçim Yöntemi | Sınıflandırma Doğruluğu | Sıralama |
|---|---|---|---|
| Mushroom | ÇOKOTUÖS-Pareto | 100 | 1 |
| Mushroom | ÇOKOTUÖS-NSGAII | 100 | 1 |
| Mushroom | CFS | 100 | 1 |
| Mushroom | CONS | 100 | 1 |

Tablo 4.16 ile Primary Tumor veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen en yüksek sınıflandırma doğrulukları ve sınıflandırma doğruluğuna göre büyükten küçüğe doğru kaçıncı sırada olduğu gösterilmiştir.

**Tablo 4.16:** En yüksek doğruluk karşılaştırması - Primary Tumor veri kümesi.

| Veri kümesi | Seçim Yöntemi | Sınıflandırma Doğruluğu | Sıralama |
|---|---|---|---|
| PrimaryTumor | ÇOKOTUÖS-NSGAII | 41,18 | 1 |
| PrimaryTumor | ÇOKOTUÖS-Pareto | 38,24 | 4 |
| PrimaryTumor | CFS | 38,24 | 4 |
| PrimaryTumor | CONS | 38,24 | 4 |

Tablo 4.17 ile Soybean veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen en yüksek sınıflandırma doğrulukları ve sınıflandırma doğruluğuna göre büyükten küçüğe doğru kaçıncı sırada olduğu gösterilmiştir.

**Tablo 4.17:** En yüksek doğruluk karşılaştırması - Soybean veri kümesi.

| Veri kümesi | Seçim Yöntemi | Sınıflandırma Doğruluğu | Sıralama |
|---|---|---|---|
| Soybean | CONS | 92,39 | 1 |
| Soybean | ÇOKOTUÖS-NSGAII | 92,39 | 1 |
| Soybean | ÇOKOTUÖS-Pareto | 91,30 | 4 |
| Soybean | CFS | 91,30 | 4 |

Tablo 4.18 ile Spect Heart veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen en yüksek sınıflandırma doğrulukları ve sınıflandırma doğruluğuna göre büyükten küçüğe doğru kaçıncı sırada olduğu gösterilmiştir.

**Tablo 4.18:** En yüksek doğruluk karşılaştırması - Spect Heart veri kümesi.

| Veri kümesi | Seçim Yöntemi | Sınıflandırma Doğruluğu | Sıralama |
|---|---|---|---|
| SpectHeart | ÇOKOTUÖS-Pareto | 81,25 | 1 |
| SpectHeart | ÇOKOTUÖS-NSGAII | 78,75 | 2 |

| | | | |
|---|---|---|---|
| SpectHeart | CONS | 78,75 | 2 |
| SpectHeart | CFS | 77,5 | 4 |

Tablo 4.19 ile Splice veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen en yüksek sınıflandırma doğrulukları ve sınıflandırma doğruluğuna göre büyükten küçüğe doğru kaçıncı sırada olduğu gösterilmiştir.

**Tablo 4.19:** En yüksek doğruluk karşılaştırması - Splice veri kümesi.

| Veri kümesi | Seçim Yöntemi | Sınıflandırma Doğruluğu | Sıralama |
|---|---|---|---|
| Splice | CFS | 99,06 | 1 |
| Splice | ÇOKOTUÖS-NSGAII | 98,75 | 2 |
| Splice | ÇOKOTUÖS-Pareto | 98,01 | 3 |
| Splice | CONS | 95,40 | 4 |

Tablo 4.20 ile Voting Records veri kümesi için, CFS, CONS, ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII ile seçilen öznitelik alt kümelerinin, 12 farklı sınıflandırma yaklaşımı uygulanarak elde edilen en yüksek sınıflandırma doğrulukları ve sınıflandırma doğruluğuna göre büyükten küçüğe doğru kaçıncı sırada olduğu gösterilmiştir.

**Tablo 4.20:** En yüksek doğruluk karşılaştırması - Voting Records veri kümesi.

| Veri kümesi | Seçim Yöntemi | Sınıflandırma Doğruluğu | Sıralama |
|---|---|---|---|
| VotingRecords | ÇOKOTUÖS-Pareto | 99,23 | 1 |
| VotingRecords | ÇOKOTUÖS-NSGAII | 99,23 | 1 |
| VotingRecords | CFS | 99,23 | 1 |
| VotingRecords | CONS | 99,23 | 1 |

## 4.4 EN YÜKSEK PERFORMANSA GÖRE SIRALAMALAR

Bu bölümde, önceki bölümdeki tablolarda görselleştirilen sonuçların gruplandırılması ve özetlenmesi ile elde edilen sonuçların yorumlanması amaçlanmıştır.

Tablo 4.11-4.20'de belirtilen en yüksek doğruluk değerleri Tablo 4.21'de toplu olarak veri kümesi bazında özetlenmiştir. Veri kümesi bazında elde edilen en yüksek doğruluk değeri koyu renk olarak ifade edilmiştir.

**Tablo 4.21:** En yüksek doğruluk sıralamaları.

| Veri kümesi | CFS | CONS | ÇOKOTUÖS-NSGAII | ÇOKOTUÖS-Pareto |
|---|---|---|---|---|
| Lymphography | 88,64 | **93,18** | 88,64 | 88,64 |
| Mushroom | **100,00** | **100,00** | **100,00** | **100,00** |
| SpectHeart | 77,50 | 78,75 | 78,75 | **81,25** |
| PrimaryTumor | 38,24 | 38,24 | **41,18** | 38,24 |
| Breast | 72,09 | 73,26 | **74,42** | 73,26 |
| Splice | **99,06** | 95,40 | 98,75 | 98,01 |
| Audiology | 79,41 | **83,82** | 82,35 | 79,41 |
| VotingRecords | **99,23** | **99,23** | **99,23** | **99,23** |
| Chess | 99,06 | **99,27** | 99,06 | **99,27** |
| Soybean | 91,30 | **92,39** | **92,39** | 91,30 |

Tablo 4.22, Tablo 4.21'deki doğruluk değerlerinin büyükten küçüğe sıralanmasıyla elde edilen başarı sıralamasını göstermektedir.

**Tablo 4.22:** En yüksek sınıflandırma doğruluğu sırasına göre sıralamalar.

| Veri kümesi | CFS | CONS | ÇOKOTUÖS-NSGAII | ÇOKOTUÖS-Pareto |
|---|---|---|---|---|
| Lymphography | 4 | **1** | 4 | 4 |
| Mushroom | **1** | **1** | **1** | **1** |

| | | | | |
|---|---|---|---|---|
| SpectHeart | 4 | 2 | 2 | **1** |
| PrimaryTumor | 4 | 4 | **1** | 4 |
| Breast | 4 | 2 | **1** | 2 |
| Splice | **1** | 4 | 2 | 3 |
| Audiology | 4 | **1** | 2 | 4 |
| VotingRecords | **1** | **1** | **1** | **1** |
| Chess | 4 | **1** | 4 | **1** |
| Soybean | 4 | **1** | **1** | 4 |

Tablo 4.22 üzerinde veri kümesi bazında 1. ve 4. sırada yer alan öznitelik seçim sonuçları, Tablo 4.23 üzerinde en iyi ve en kötü sonuçların sayısı olarak gösterilmektedir.

**Tablo 4.23:** En iyi ve en kötü sıralama elde etme adetleri.

| Yöntem | En İyi | En Kötü |
|---|---|---|
| CFS | 3 | 7 |
| CONS | 6 | 2 |
| ÇOKOTUÖS-NSGAII | 5 | 2 |
| ÇOKOTUÖS-Pareto | 4 | 4 |

Tablo 4.22'de tek başına ilk sırayı alarak diğer üç yaklaşımdan daha yüksek doğruluk sağlayan öznitelik seçim yaklaşımları Tablo 4.24'te gösterilmiştir.

**Tablo 4.24:**.Veri kümesine göre diğer yaklaşımları domine etme sayıları.

| Yöntem | Domine Etme Sayısı | Veri kümeleri |
|---|---|---|
| CFS | 1 | Splice |
| Consistency | 2 | Lymphography, Audiology |

| | | |
|---|---|---|
| ÇOKOTUÖS-NSGAII | 2 | PrimaryTumor, Breast |
| ÇOKOTUÖS-Pareto | 1 | SpectHeart |

CONS ve CFS yaklaşımlarını ayrı ayrı uygulamaya nispeten, hem bu iki yaklaşımlarla merit değerlerinin hesaplanması hem de optimizasyon işlemlerinin ve sarmalayıcı adımlarının takip edilmesinden dolayı önerilen yaklaşımların gerekli işlem süresi ve sistem kaynağı tarafında daha maliyetli olacağı aşikardır. Rastgele orman algoritmasının C4.5 karar ağacına göre daha yüksek işlem yükü ve analiz süresi gerektirmesine rağmen daha yüksek sınıflandırma doğruluğu sağlayabiliyor olması nedeni ile yaygın kullanılması motivasyonuna paralel olarak bu çalışmada önerilen ÇOKOTUÖS yaklaşımlarının, CFS ve CONS yaklaşımlarından daha yüksek maliyetli ve daha uzun süre ihtiyacı duyacağı kabul edilerek ilerlenmiş ve bu nedenle aşikar olan algoritmaların çalışma sürelerinin tartışılması kapsam dışında bırakılmıştır. Bu nedenle, çıktıların değerlendirilmesi aşamasında sınıflandırma doğruluğu ve F1 skor değerleri özetlenmiştir. Bu çalışmada hedef, daha yüksek sınıflandırma doğruluğu değerine ulaşabilmek olduğu için, on farklı veri kümesi üzerinde dört farklı öznitelik seçim yaklaşımı ve kullanılmış olan 12 farklı sınıflandırma algoritması ile üretilen 480 adet sınıflandırma tahmin modelinin ürettiği tahminler üzerinde en yüksek sınıflandırma doğruluğu değerleri özetlenmiş ve bu sınıflandırma doğruluğu değerinin yüksekliğine göre yaklaşımlar ile elde edilen sonuçlar sıralanmıştır.

Sonuçlar detaylı olarak incelendiğinde, yapılan çalışmalarda elde edilen sınıflandırma doğruluğu değerlerinin ortalamaları üzerinden anlamlı farklılıkların ortaya çıkmadığı görülmüştür. Dolayısıyla, sınıflandırma doğruluğu ortalamaları üzerinden elde edilen sonuçların yorumlanmasının pek yol gösterici olmadığı görülmektedir. Bu nedenle, en yüksek performans kıyaslamaları göz önünde bulundurulmuştur.

F1 skor ortalamaları incelendiğinde, ÇOKOTUÖS-NSGA II, Audiology ve Spect Heart veri kümelerinde en iyi dereceleri almasının yanı sıra Breast ve Lymphography veri kümelerinde en kötü dereceleri almıştır. ÇOKOTUÖS-Pareto ise Mushroom, Breast veri kümelerinde en iyi dereceleri, Spect Heart, Chess, Voting Records ve Primary Tumor veri kümelerinde en kötü dereceleri almıştır.

## 4.5 F1 SKOR STANDART SAPMA KARŞILAŞTIRMALARI

Tablo 4.25 ile on veri kümesi üzerinde gerçekleştirilen 12 sınıflandırma çalışmasından elde edilen F1 skorların standart sapma değerlerine göre en düşük ve en yüksek değerleri sağlayan öznitelik seçim algoritmaları gösterilmiştir.

**Tablo 4.25:** F1 Skor standart sapmalarına göre karşılaştırmalar.

| Veri Kümesi | En Yüksek F1 Skor Standart Sapması | En Düşük F1 Skor Standart Sapması |
|---|---|---|
| Splice | CONS | CFS |
| Audilogy | CONS | MOCOCOFS-NSGAII |
| Lymphography | CONS | MOCOCOFS-NSGAII |
| Soybean | CONS | MOCOCOFS-NSGAII |
| Breast | CONS | MOCOCOFS-Pareto |
| Mushroom | CONS | MOCOCOFS-Pareto |
| Primary Tumor | MOCOCOFS-NSGAII | CFS |
| Voting Records | MOCOCOFS-Pareto | CFS |
| Chess | MOCOCOFS-Pareto | CFS |
| SpectHeart | MOCOCOFS-Pareto | MOCOCOFS-NSGAII |

Tablo 4.25'e göre üzerinde çalışılan on veri kümesinden altı tanesinde CONS ile, dört tanesinde MOCOCOFS-Pareto ile seçilen öznitelik alt kümeleri ile gerçekleştirilen sınıflandırma çalışmalarından elde edilen F1 skor değerlerinin standart sapmalarının diğer uygulanan yöntemlere göre nispeten daha yüksek seviyede olduğu görülürken beş veri kümesinde MOCOCOFS-NSGAII, dört veri kümesinde ise CFS kullanıldığında en düşük standart sapma değerlerinin gözlemlendiği söylenebilir. F1 skorların standart sapma değerlerinin incelendiği bu bakış açısı ile, bu tezde gerçekleştirilen CONS ve MOCOCOFS-NSGAII uygulamalarının, CFS ve MOCOCOFS-Pareto uygulamalarına göre nispeten daha yüksek standart sapmaya sahip olduğu, sınıflandırma algoritmalarının performanslarının daha çok dalgalanma gösterdiği söylenebilir.

# 5. TARTIŞMA VE SONUÇ

Bu tez çalışması kapsamında, sınıflandırma görevine yönelik olarak literatürde bulunan öznitelik alt kümesi seçimi yaklaşımlarına alternatif bir hibrit çok amaçlı optimizasyon destekli yeni iki yaklaşımın önerilmesi çalışmaları gerçekleştirilmiştir. Öncelikli olarak korelasyon (CFS) ve tutarlılık (CONS) merit değerlerinin oluşturduğu kartezyen uzay üzerinde ideal öznitelik alt kümesinin aranması önerilmiştir. Tanımlanan kartezyen uzay üzerinde Pareto ve NSGA-II optimizasyon algoritmaları kullanılarak CFS ve CONS merit değerlerini maksimize etme amacına yönelik olarak aday öznitelik alt küme sayısınının düşürülmesi sağlanmıştır. Düşük sayıda bulunan adayların her biri için sarmalayıcı yaklaşım uygulanarak, bu adaylar arasından sınıflandırma doğruluğunu en yüksek seviyede sağlayabilecek adayın seçilmesi önerilmiştir. CONS ve CFS merit değerlerinin hesaplanması aşamasında, veri kümesi içerisindeki sürekli öznitelikler için bir kesikleştirme işlemi uygulanarak tüm özniteliklerin kategorik tipte olması sağlandığı için, kategorik özniteliklerle yüksek performasla çalışan güncel algoritmalardan biri olan CatBoost algoritması sarmalayıcı yaklaşım aşamasında kullanılmıştır. Bu işleyişin tamamına Çok Amaçlı Korelasyon ve Tutarlılık Bazlı Öznitelik Alt Kümesi Seçimi (ÇOKOTUÖS) adı verilerek literatüre yeni bir yaklaşım olarak kazandırılmıştır.

Bu tez çalışmasında önerilen ÇOKOTUÖS yaklaşımı, UCI veri deposundan temin edilen 10 veri kümesi üzerinde CFS ve CONS yaklaşımları ile birlikte uygulanarak her veri kümesi için veri kümesinin içerdiği gürültüyü azaltmaya yönelik olarak dört farklı öznitelik seçimi sonucu elde edilmiştir. Elde edilen öznitelik alt kümelerinin sonuçlarını karşılaştırmak için 12 farklı sınıflandırma algoritması ile elde edilen tahmin sonuçları karşılaştırmalı olarak incelenmiştir.

ÇOKOTUÖS çalışmaları kapsamında hesaplanan CFS ve CONS merit değerlerinin kartezyen düzlem üzerindeki dağılımları incelendiğinde genel olarak iki farklı tipte dağılımın gözlemlendiği söylenebilir.

Birinci tipteki dağılımlarda, aşikar bir Pareto küme elde edilebilmektedir. Breast, Chess, Splice, SpectHeart, Mushroom, Voting Records veri kümeleri için Pareto optimizasyon yaklaşımı ile, sarmalayıcı aşamasında değerlendirilebilecek kadar aday üretilebilmiştir ve alternatif sonuçlar da ortaya konulmuştur. Bu uygulamalarda, NSGA-II yaklaşımı ile elde edilen adaylar da sarmalayıcı aşamasında değerlendirilebilecek sayıda olup, genel itibari ile Pareto yaklaşımına göre daha az adayı sarmalayıcı aşamasına taşımaktadır.

İkinci tipteki dağılımlarda ise, geniş bir Pareto küme elde edilememektedir. Pareto yaklaşımı ile geniş bir Pareto küme elde edilemediği durumda, sarmalayıcı aşaması ile değerlendirilmek üzere anlamlı sayıda adayın bulunamaması bir handikap oluşturmaktadır. Audiology, Lymphography, Primary Tumor, Soybean veri kümeleri için bu durum gözlemlenmiştir. Bu veri kümelerinde, Pareto optimizasyonu kullanılarak CFS ve CONS için iyi birer alternatif bulunamamıştır. Ancak bu veri kümeleri için NSGA-II optimizasyonu kullanılarak oluşturulan alternatiflerde sarmalayıcı uygulaması ile değerlendirilebilecek adaylar her zaman bulunabilmiş ve CFS ve CONS uygulamaları tarafından seçilmemiş farklı öznitelik alt kümesi adayları önerilebilmiştir.

Bu ayrıma göre incelendiğinde, Pareto kümenin değerlendirilebilecek kadar genişlediği durumlarda, NSGA-II yaklaşımının genel olarak Pareto yaklaşımından daha düşük sarmalayıcı eforu ihtiyacı ile sonuç üretme şansı olduğu görülmektedir. Pareto yaklaşımı ile, az sayıda öznitelik alt kümesi adayı ortaya çıktığı durumlarda alternatif üretilememesine karşın, NSGA-II ile farklı alternatiflerin sarmalayıcı aşamaya taşınabilmesi sağlanmıştır. Pareto kümenin geniş olmadığı dağılımlarda, ÇOKOTUÖS-Pareto yaklaşımı içerisindeki sarmalayıcı aşamaya fazla aday taşınamadığı için, sarmalayıcı aşamanın sınıflandırma doğruluğu değerinin yüksekliğine olan katkısından yeterince yararlanılamamaktadır. Buna karşın, ÇOKOTUÖS-NSGAII, Pareto kümenin geniş olmadığı durumlarda bile sarmalayıcı aşamaya ÇOKOTUÖS-Pareto yaklaşımına göre nispeten daha fazla aday taşıyabilmiş ve sarmalayıcı aşamanın sınıflandırma doğruluğu değerinin yüksekliğini sağlama katkısından istifade edebilmiştir.

Sonuç olarak,

CFS yaklaşımının Splice,

CONS yaklaşımının Lymphography, Audiology,

ÇOKOTUÖS-NSGAII yaklaşımının PrimaryTumor, Breast ve

ÇOKOTUÖS-Pareto yaklaşımının SpectHeart

için bu çalışmada kullanılan diğer yaklaşımlardan daha yüksek sınıflandırma doğruluğu değeri elde edilebilen öznitelik alt kümelerinin seçilmesini sağladığı görülmüştür.

EK-2 ile gösterildiği şekilde, on farklı veri kümesi üzerinde, 12 farklı sınıflandırma ile gerçekleştirilen sınıflandırma çalışmalarında Lymphography veri kümesi üzerinde en iyi sonucu Lojistik Regresyon, diğer veri kümeleri üzerinde ise ağaç temelli CatBoost, Rasgele Orman, LGBM ve Ekstra Ağaçlar algoritmalarının kullanılması ile en iyi sonuçların elde edilebildiği görülmektedir.

Literatürde bulunan güncel öznitelik seçimi yaklaşımı öneren çalışmalarda, kullanılan öznitelik seçimi adetleri, sınıflandırıcı adetleri, veri kümesi sayıları ve çıktıların incelenmesine yönelik olarak gerçekleştirilen uygulamalar Tablo 5.1'de karşılaştırılmıştır. Çalışmalar üzerinde yapılan incelemelerde sınıflandırma için öznitelik seçimi çalışmalarında kullanılan belirgin bir karşılaştırma yaklaşımının olmadığı ve en yüksek sınıflandırma doğruluğu, F1 skor, ortalamalar ve standart sapmaların yaygın olarak kullanıldığı görülmüştür. İncelenen diğer çalışmalarda bir ila sekiz arasında değişen sayıda sınıflandırıcı kullanıldığı görülmektedir. Bu çalışmada, aralarında CatBoost, LightGBM ve ExtraTrees gibi güncel algoritmaların da bulunduğu 12 farklı sınıflandırıcının kullanılması ile literatürdeki çalışmalara nispeten çeşitliliğin ve güncelliğin sağlanması açısından önemli olduğu söylenebilir. İncelenen diğer çalışmalarda bir ila 10 arasında değişen sayıda öznitelik seçimi yaklaşımı ve bir ila 18 arasında değişen sayıda veri kümesi kullanıldığı görülmüştür. Bu tez çalışmasında kullanılan öznitelik seçimi yaklaşımı sayısı ve veri kümesi sayısı karşılaştırılan diğer çalışmalardaki sayı aralıklarının içerisinde kalmaktadır.

**Tablo 5.1:** Literatürdeki çalışmaların uygulama ve sonuç karşılaştırmaları.

| Çalışma | Seçim yaklaşımı Sayısı | Sınıflandırıcı Sayısı | Veri Kümesi Sayısı | Değerlendirme yaklaşımı |
|---|---|---|---|---|
| Jimenez ve diğ. (2022) | 8 | 8 | 5 | Öznitelikleri değerlendirme metriğine göre sıraya dizdikten sonra ilk 50, 100 gibi değişik adetlerde özniteliği seçerek farklı sınıflandırma algoritmalarından elde edilen en yüksek sınıflandırma |

| | | | | |
|---|---|---|---|---|
| | | | | doğruluğuna göre incelemede bulunmuşlardır. |
| Al-Tashi ve diğ. (2019) | 5 | 1 | 18 | Kullandıkları öznitelik seçimi yaklaşımının parametrelerini değiştirerek tekrarlı uygulamalardan elde ettikleri öznitelik alt kümelerine KNN uygulayarak doğruluk, uyum (fitness), öznitelik adedi ve süre sonuçlarının ortalamalarını değerlendirmişlerdir. |
| Alia ve Taweel (2021) | 4 | 2 | 16 | Önerdikleri öznitelik seçimi yöntemi ve literatürde bulunan 3 farklı öznitelik seçimi yöntemi ile seçilen öznitelik al Karar ağacı ve naive bayes sınıflandırıcı ile elde edilen sınıflandırma doğruluğu değerlerini karşılaştırmışlardır. |
| Assafo ve Langendoerfer (2021) | 4 | 1 | 1 | Dört farklı öznitelik seçimi yaklaşımı ile elde edilen öznitelik alt kümeleri üzerinde sınıflandırma doğruluğu, F1-skor, pozitif doğruluk oranı(true pozitif rate) ve pozitif yanlışlık oranı (false positive rate) hesaplamaları karşılaştırmaları yapılmıştır ayrıca sınıflandırma doğruluğuna ait ortalama, standart sapma, minimum, maksimum |

| | | | | |
|---|---|---|---|---|
| | | | | değerleri karşılaştırmalı olarak incelenmiştir. |
| Albashah ve Rais (2022) | 3 | 1 | 14 | Üç farklı yapay sinir ağlarını sarmalayıcı olarak kullanan öznitelik seçimi yaklaşımlarının iteratif olarak kullanımı ile seçilen öznitelik alt kümelerinin barındırdığı öznitelik sayılarının minimum, maksimum, ortalama, standart sapma ve medyan değerlerini karşılaştırmışlardır. |
| Piri ve diğ. (2022a) | 4 | 5 | 10 | Dokuz farklı UCI platformundan alınan veri kümesi ve bir adet COVID-19 veri kümesi üzerinde dört farklı öznitelik seçimi yaklaşımı üzerinde ortalama doğruluk, ortalama boyut ve ortalama süre karşılaştırması yapmışlardır. |
| Ghanem ve diğ. (2022) | 2 | 1 | 5 | Beş farklı siber saldırı veri kümesi üzerinde iki farklı öznitelik seçimi yaklaşımını yapay sinir ağları ile birlikte kullanarak sınıflandırma doğruluğu, pozitif doğruluk oranı(true pozitif rate) ve pozitif yanlışlık oranı (false positive rate) üzerinden karşılaştırma yapmışlardır. |

| | | | | |
|---|---|---|---|---|
| Zhang ve diğ. (2022) | 10 | 1 | 16 | 16 farklı veri kümesi üzerinde, tek amaçlı ve çok amaçlı optimizasyon yaklaşımlarını barındıran 10 farklı öznitelik alt kümesi seçimi yaklaşımından elde edilen öznitelik alt kümelerine KNN sınıflandırıcı uygulanarak sınıflandırma hata oranları karşılaştırılmıştır. |
| Ghaleb ve diğ. (2022) | 2 | 1 | 3 | Üç farklı siber saldırı veri kümesi üzerinde iki farklı öznitelik seçimi uygulamasından elde edilen öznitelik alt kümelerine yapay sinir ağlarının uygulanması sonrası pozitif doğruluk oranı(true pozitif rate) ve pozitif yanlışlık oranı (false positive rate) üzerinden karşılaştırma yapmışlardır. |
| Xue ve diğ. (2022) | 5 | 1 | 6 | Altı farklı veri kümesi üzerinde, beş farklı öznitelik alt kümesi seçimi yöntemi ile elde edilen öznitelik alt kümelerinin ters nesil mesafesi (inverted generational distance) ve hiper hacim metriği (hyper-volume metric) değerlerinin ortalamalarının ve standart sapmalarının karşılaştırılması yapılmıştır. |

| | | | | |
|---|---|---|---|---|
| Assafo ve Langendoerfer (2021) | 4 | 1 | 1 | Dört farklı öznitelik seçimi yaklaşımı ile elde edilen öznitelik alt kümeleri üzerinde sınıflandırma doğruluğu, F1-skor, pozitif doğruluk oranı(true pozitif rate) ve pozitif yanlışlık oranı (false positive rate) hesaplamaları karşılaştırmaları yapılmıştır ayrıca sınıflandırma doğruluğuna ait ortalama, standart sapma, minimum, maksimum değerleri karşılaştırmalı olarak incelenmiştir. |
| Bu tez çalışması (Sezer 2023) | 4 | 12 | 10 | On farklı veri kümesi üzerinde, dört farklı öznitelik seçim yaklaşımı ile seçilen öznitelik alt kümelerine 12 farklı sınıflandırıcı uygulanarak sınıflandırma doğruluğu ve F1-skor üzerinden en yükseğe göre sıralama, ortalama ve standart sapma değerleri üzerinden karşılaştırmalar yapılmıştır. |

Xu ve diğ. (2018), Abu Shanab ve Khoshgoftaar (2018), Azarmdel ve diğ. (2020), Algin ve diğ. (2020) ve Jimenez ve diğ. (2022) tarafından gerçekleştirilen çalışmalarda CFS ve CONS öznitelik seçimleri, farklı varyasyonlarla kullanılmış ancak bu çalışmaların hiçbirinde öznitelik alt kümelerini değerlendirmek için CFS ve CONS yaklaşımları birlikte kullanılmamıştır. Jimenez ve diğ. (2022) gerçekleştirdikleri çalışmada, skorlanan öznitelik alt kümelerini eleman sayılarını minimize etme amacı ile birlikte ikili korelasyon ve ikili tutarlılık merit değerlerini kullanmışlardır. Ancak Jimenez ve diğ. (2022) çalışmalarını gen veri kümeleri üzerinde

gerçekleştirdikleri için, bu tez çalışması kapsamında gerçekleştirilen çalışmalar ile performans karşılaştırması yapılamamıştır.

Öznitelik alt kümesi seçim çalışmalarında, genellikle birkaç farklı algoritma ve arama stratejilerinden faydalanılarak oluşturulan yöntemler ile elde edilen öznitelik alt kümelerinin karşılaştırmasının (*benchmark*) yapıldığı görülmektedir. Elde edilen sonuçlara dayanarak, araştırmacıların ÇOKOTUÖS-Pareto ve ÇOKOTUÖS-NSGAII yaklaşımlarını karşılaştırma çalışmalarına dahil etmeleri durumunda alternatif etkin öznitelik alt kümeleri elde edilebileceği görülmüştür.

Makine öğrenmesi yaklaşımları incelendiğinde korelasyon metrikleri ile çalışan çok etkili algoritmaların varlığından bahsetmek mümkün değildir. Öte yandan, kategorik verilerdeki tutarlılığın hesaplanması, kategorik veri kümesinin kodlanmış versiyonunda KNN algoritmasının uygulanması ile örtüşmektedir. Bu nedenle, tutarlılık metriğinin korelasyona göre daha iyi açıklayıcı olabileceği ve korelasyonun tek başına çok güçlü bir açıklayıcı olmadığı söylenebilir. Bu nedenle, bu tez çalışmasında önerilen yaklaşımda olduğu gibi tutarlılık tabanlı öznitelik seçimi ve sarmalayıcı bir yaklaşımla desteklenen korelasyon tabanlı öznitelik seçiminin kullanılmasının daha faydalı olabilmektedir.

MOCOCOFS yaklaşımı, öznitelik alt kümesi değerlendirme için hesaplanan korelasyon ve tutarlılık merit değerlerinin birlikte oluşturduğu kartezyen düzlem üzerinde, sarmalayıcı yaklaşım ile en iyi adayın aranması yaklaşımına dayanır. Bu arayış için, Pareto ve NSGA-II yaklaşımları bu tez çalışması kapsamında uygulanmıştır. Farklı optimizasyon yaklaşımları ile de bu kartezyen düzlem üzerinde aday arayışı yapılabilir.

Bu tez çalışmasında, Breast veri kümesi için tam arama (*exhaustive search*), diğer veri kümeleri için rastgele arama (*random search*) uygulanarak korelasyon ve tutarlılık merit değerlerine dayalı kartezyen düzlem oluşturulmuştur. Sezgisel arama (*heuristic search*) gibi farklı yaklaşımlar kullanılarak da bu iki merit değerinin aynı anda değerlendirildiği yaklaşımlar geliştirilebilir.

Dağılımlar üzerinden inceleme yapıldığında, genel olarak iki farklı tipte dağılımın gözlemlendiği söylenebilir. Bu durumun yaşanmasının sebebi olarak, bazı veri kümelerinin aşırı yüksek CONS merit değerlerine sahip olması ve buna bağlı olarak eğrisel bir Pareto

kümenin oluşmaması söylenebilir. Bu durum, hazır veri kümeleri ile çalışmış olmaktan kaynaklı olabilir.

Al-Tashi ve diğ. (2019), öznitelik seçimi için hibrit gri kurt optimizasyonu kullanarak ikili optimizasyon algoritması (BGWOPSO), öznitelik seçimi için benzetilmiş tavlama ile hibrit balina optimizasyon algoritması (WOASAT-2), ikili gri kurt optimizasyonu-2 (bGWO2), ikili parçacık sürü optimizasyonu (BPSO), ikili genetik algoritma (*Binary genetic algorithm* - BGA) kullanarak Voting Records, Spect Heart ve Lymphography veri kümeleri üzerinde sınıflandırmaya yönelik öznitelik seçimi çalışmaları yapmışlardır. Voting Records için elde ettikleri sınıflandırma doğruluğu değerleri BGWOPSO için %98, WOASAT-2 için %93, bGWO2 için %94, BPSO için %94 ve BGA için %98 şeklindedir. Bu tez çalışmasında kullanılan ÇOKOTUÖS-Pareto için %99,23 ve ÇOKOTUÖS-NSGAII için de %99,23 sınıflandırma başarısı yakalanarak Al-Tashi ve diğerlerinin (2019) tüm denemelerinden daha iyi sonuçlara ulaşıldığı görülmektedir. Al-Tashi ve diğerlerinin (2019) SpectHeart veri kümesi üzerinde elde ettikleri sınıflandırma başarıları BGWOPSO için %88, WOASAT-2 için %82, bGWO2 için %77, BPSO için %78, BGA için %88 şeklindedir. Bu tez çalışmasında SpectHeart veri kümesi için yapılan analizlerde ÇOKOTUÖS-Pareto için %78,75 ve ÇOKOTUÖS-NSGAII için %81,25 sınıflandırma başarısı elde edilmiştir. Al-Tashi ve diğ. (2019)'nin Lymphography veri kümesi üzerinde elde ettikleri sınıflandırma başarıları BGWOPSO için %92, WOASAT-2 için %89, bGWO2 için %70, BPSO için %69 ve BGA için %71 şeklindedir. Bu tez çalışmasında Lymphography veri kümesi için yapılan analizlerde ÇOKOTUÖS-Pareto için %88,64 ve ÇOKOTUÖS-NSGAII için %88,64 sınıflandırma başarısı elde edilmiştir. Bu tez çalışmasında SpectHeart ve Lymphography veri kümeleri üzerinde elde edilen sonuçlar Al-Tashi ve diğerlerinin (2019) uygulamalarının bazılarından daha iyi sonuç vermiş olsa da, Al-Tashi ve diğerlerinin (2019) elde ettiği yüksek sınıflandırma doğruluklarına ulaşabilmişlerdir. Al-Tashi ve diğerlerinin (2019) çalışmaları ile bu tezde gerçekleştirilen çalışmalar karşılaştırıldığında ortak kullanılan Voting Records, SpectHeart ve Lymphography veri kümelerinden sadece Voting Records üzerinde Al-Tashi ve diğerlerinden (2019) daha iyi sonuçlar elde edildiği görülmektedir. SpectHeart ve Lymphography veri kümeleri üzerinde daha düşük başarı elde edilmesi, bu tez çalışması kapsamında alternatif öznitelik alt kümesi uzayı içerisinden sadece 500 adet aday öznitelik alt kümesinin rastgele seçilerek yapılmış olması olabilir. Daha geniş bir değerlendirme alt uzayı seçildiğinde bu tezde önerilen yaklaşımların daha iyi sonuçlar üretebileceği düşünülmektedir.

Bu tez çalışmasında önerilen yaklaşımın, az sayıda adaya sarmalayıcı yaklaşımın uygulanmasını sağlaması sınıflandırma başarısını yüksek tutmayı sağlayan önemli bir fonksiyondur. Usman ve diğ. (2020), Z. F. Zhang ve diğ. (2021), Alia ve Taweel (2021) tarafından önerilen filtreleyici yaklaşımlar bir sarmalayıcı barındırmaması nedeni ile yüksek sınıflandırma başarısını teyit eden bir fonksiyon içermemektedir. Bu açıdan önerilen ÇOKOTUÖS yaklaşımının, belirtilen filtreleyici yaklaşımlardan ve benzerlerinden farklı olarak sınıflandırma başarısını yüksek tutmaya yönelik bir sarmalayıcı adımı barındırdığı için, diğer yaklaşımlara nispeten daha iyi sonuçlar üretmesinin beklenebileceği söylenebilir.

Özmen (2016), Ghanem ve diğ. (2022) ve Piri ve diğ. (2022) tarafından önerilen yaklaşımlarda ise sarmalayıcı aşamada oldukça fazla adayın değerlendirilmesi gerekmektedir ve bu ciddi bir işlem yükü getirmektedir. Önerilen ÇOKOTUÖS yaklaşımının, sarmalayıcı aşama öncesinde öznitelik alt kümelerini skorlaması ve ardından optimizasyon yaklaşımlarından yararlanarak adayları oldukça aza indirgemesi ile sarmalayıcı aşamada oldukça düşük bir işlem yükü söz konusudur. Dolayısıyla, ÇOKOTUÖS yaklaşımı, bu araştırmacıların önermiş oldukları yaklaşımlardan sarmalayıcı aşamada gereken işlem yüküne göre daha iyi bir seçenek sunmaktadır.

Gelecek çalışmalarda tez kapsamında önerilen yaklaşımın Python ve R programlama dillerinde ÇOKOTUÖS öznitelik seçimi kütüphanelerinin oluşturulması ve WEKA platformuna entegre edilmesi planlanmaktadır. Bu sayede, tez kapsamında önerilen ÇOKOTUÖS yaklaşımı diğer araştırmacılar tarafından kolaylıkla uygulanabilir halde sunulacaktır. Gelecek çalışmalarda, tez kapsamında önerilen yaklaşımın Python ve R programlama dillerinde ÇOKOTUÖS öznitelik seçimi kütüphanelerinin oluşturulması ve WEKA platformuna entegre edilmesi planlanmaktadır. Bu sayede tez kapsamında önerilen ÇOKOTUÖS yaklaşımı diğer araştırmacılar tarafından kolaylıkla uygulanabilir halde sunulacaktır. Bu çalışmada, literatürde yeni bir öznitelik seçimi yaklaşımı öneren diğer tüm araştırmacılar gibi yaygın kullanılan herkese açık veri kümeleri kullanılmıştır. Gelecek çalışmalarda daha geniş veri kümeleri üzerinde bu çalışmaların gerçekleştirilmesi ve sonuçlarının incelenmesi, yöntemin başarısını değerlendirmek faydalı olacaktır. Gelecek çalışmalarda, sınıflandırma görevleri için öznitelik seçimi için daha farklı kartezyen uzaylar tanımlanabilir ve bu tez çalışmasına benzer şekilde Pareto ve NSGA-II algoritmaları, tanımlanan yeni uzaylarda kullanılabilir. Tutarlılık ölçüsünün aslında KNN algoritması ile aynı hesaplamaya denk geldiğinin bu çalışma kapsamında ilk kez

ortaya atılmasına paralel olarak, literatürde bulunan diğer ölçüler karşılaştırmalı olarak incelenebilir.

# KAYNAKLAR


Abraham, R., Simha, J. B., and Iyengar, S. S. (2009). Effective Discretization and Hybrid feature selection using Naïve Bayesian classifier for Medical data mining. *International Journal of Computational Intelligence Research*, *5*(2), 116-129.

Abu Shanab, A., and Khoshgoftaar, T. M. (2018). Filter-based subset selection for easy, moderate, and hard bioinformatics data. *2018 IEEE 19th International Conference on Information Reuse and Integration for Data Science, IRI 2018*, 372-377. Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/IRI.2018.00062

Afshari, H., Hare, W., and Tesfamariam, S. (2019). Constrained multi-objective optimization algorithms: Review and comparison with application in reinforced concrete structures. *Applied Soft Computing Journal*, *83*, 1-18. https://doi.org/10.1016/J.ASOC.2019.105631

Agrawal, P., Abutarboush, H. F., Ganesh, T., and Mohamed, A. W. (2021). Metaheuristic Algorithms on Feature Selection: A Survey of One Decade of Research (2009-2019). *IEEE ACCESS*, *9*, 26766-26791. https://doi.org/10.1109/ACCESS.2021.3056407

Akbulut, D. (2019). *Optimization approaches for classification and feature selection using overlapping hyperboxes* (Yüksek Lisans). Orta Doğu Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.

Akhtar, E. S. F. (2017). *Practical Reinforcement Learning: Develop self-evolving, intelligent agents with OpenAI Gym, Python and Java*. Packt publishing. ISBN: 1787128725

Akpınar, H. (2014). *Data: Veri madenciliği Veri analizi*. Papatya yayıncılık, İstanbul.

Albashah, N. L. S., and Rais, H. M. (2022). Population Initialization Factor in Binary Multi-Objective Grey Wolf Optimization for Features Selection. *IEEE ACCESS*, *10*, 114942-114958. https://doi.org/10.1109/ACCESS.2022.3218056

Algin, R., Alkaya, A. F., and Agaoglu, M. (2020). Feature selection via computational intelligence techniques. *Journal of Intelligent & Fuzzy Systems*, *39*(5), 6205-6216. https://doi.org/10.3233/JIFS-189090

Alhaj, T. A., Siraj, M. M., Zainal, A., Elshoush, H. T., ve Elhaj, F. (2016). Feature selection using information gain for improved structural-based alert correlation. *PLoS ONE*, *11*(11). https://doi.org/10.1371/journal.pone.0166017

Alia, A., and Taweel, A. (2021). Enhanced Binary Cuckoo Search With Frequent Values and Rough Set Theory for Feature Selection. *IEEE ACCESS*, *9*, 119430-119453. https://doi.org/10.1109/ACCESS.2021.3107901

Aljarah, I., Faris, H., Heidari, A. A., Mafarja, M. M., Al-Zoubi, A. M., Castillo, P. A., and Merelo, J. J. (2021). A Robust Multi-Objective Feature Selection Model Based on Local Neighborhood Multi-Verse Optimization. *IEEE ACCESS*, *9*, 100009-100028. https://doi.org/10.1109/ACCESS.2021.3097206

Almuallim, H., and Dietterich, T. G. (1991). Learning With Many Irrelevant Features. *Association for the Advancement of Artificial Intelligence*, 547-552.

Alpaydin, E. (2014). *Introduction to machine learning*, 3rd ed., The MIT Press, London.

Al-Tashi, Q., Abdulkadir, S. J., Rais, H. M., Mirjalili, S., and Alhussian, H. (2020). Approaches to multi-objective feature selection: A systematic literature review. *IEEE Access* , *8*, 125076-125096.

Al-Tashi, Q., Abdulkadir, S. J., Rais, H. M., Mirjalili, S., Alhussian, H., Ragab, M. G., and Alqushaibi, A. (2020). Binary Multi-Objective Grey Wolf Optimizer for Feature Selection in Classification. *IEEE ACCESS*, *8*, 106247-106263. https://doi.org/10.1109/ACCESS.2020.3000040

Al-Tashi, Q., Kadir, S. J. A., Rais, H. M., Mirjalili, S., and Alhussian, H. (2019). Binary Optimization Using Hybrid Grey Wolf Optimization for Feature Selection. *IEEE ACCESS*, *7*, 39496-39508. https://doi.org/10.1109/ACCESS.2019.2906757

Al-Thanoon, N. A., Algamal, Z. Y., and Qasim, O. S. (2021). Feature selection based on a crow search algorithm for big data classification. *Chemometrics and Intelligent Laboratory Systems*, *212*. https://doi.org/10.1016/j.chemolab.2021.104288

Assafo, M., and Langendoerfer, P. (2021). A TOPSIS-Assisted Feature Selection Scheme and SOM-Based Anomaly Detection for Milling Tools Under Different Operating Conditions. *IEEE ACCESS*, *9*, 90011-90028. https://doi.org/10.1109/ACCESS.2021.3091476

Asthana, P., and Hazela, B. (2020). Applications of machine learning in improving learning environment. *Multimedia Big Data Computing for IoT Applications: Concepts, Paradigms and Solutions*, 417-433.

Azarmdel, H., Jahanbakhshi, A., Saeid Mohtasebi, S., and Rosado Muñoz, A. (2020). Evaluation of image processing technique as an expert system in mulberry fruit grading based on ripeness level using artificial neural networks (ANNs) and support vector machine (SVM). *Postharvest Biology and Technology*, *166*, 1-12. https://doi.org/10.1016/j.postharvbio.2020.111201

Balaban, E. M., ve Kartal, E. (2018). *Veri Madenciliği ve Makine Öğrenmesi Temel Kavramlar, Algoritmaları ve R dili ile Uygulamaları*, 2nd ed., Çağlayan Kitabevi, İstanbul.

Bareiss, E. R., Porter, B. W., and Wier, C. C. (1988). Protos: An exemplar-based learning apprentice. *International Journal of Man-Machine Studies*, *29*(5), 549-561.

Barlow, H. B. (1989). Unsupervised learning. *Neural computation*, *1*(3), 295-311.

Baxter, C. W., Zhang, Q., Stanley, S. J., Shariff, R., Tupas, R.-R., and Stark, H. L. (2011). Drinking water quality and treatment: the use of artificial neural networks. *Canadian Journal of Civil Engineering*, *28*(S1), 26-35. https://doi.org/10.1139/L00-053

Bayes, T., and Price, R. (1763). An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society of London*, *53*, 370-418.

Benítez-Hidalgo, A., Nebro, A. J., García-Nieto, J., Oregi, I., and Ser, J. del. (2019). jMetalPy: A Python framework for multi-objective optimization with metaheuristics. *Swarm and Evolutionary Computation*, *51*. https://doi.org/10.1016/j.swevo.2019.100598

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer, New York.

Blank, J., and Deb, K. (2020). Pymoo: Multi-Objective Optimization in Python. *IEEE Access*, *8*, 89497-89509. https://doi.org/10.1109/ACCESS.2020.2990567

Blum, A. L., ve Langley, P. (1997). Selection of relevant features and examples in machine learning. *Artificial Intelligence*, *97*, 245-271.

Bottou, L. (1991). Stochastic gradient learning in neural networks. *Proceedings of Neuro-Nimes*, *91*(8), 1-12.

Breiman, L. (2001). Random Forests. *Machine learning*, *45*, 5-32.

Brennan, T., and Oliver, W. L. (2013). Emergence of machine learning techniques in criminology: implications of complexity in our data and in research questions. *Criminology & Public Policy*, *12*, 551.

Burke, E. K., and Kendall, G. (2014). *Search methodologies: Introductory tutorials in optimization and decision support techniques*. Springer, New York. https://doi.org/10.1007/978-1-4614-6940-7

Burkov, A. (2019). *The hundred-page machine learning book* (C. 1). Quebec City, QC, Canada. ISBN: 978-1999579500.

Cestnik, B., Konenenko, I., and Bratko, I. (1987). A knowledge-elicitation tool for sophisticated users. *Progress in Machine Learning*, 31-45.

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., and Wirth, R. (2000). CRISP-DM 1.0: Step-by-step data mining guide. *SPSS inc*, *9*(13), 1-73.

Cover, T. M., and Hart, P. E. (1967). Nearest Neighbor Pattern Classification. *IEEE Transactions on Information Theory*, *13*(1), 21-27. https://doi.org/10.1109/TIT.1967.1053964

Cui, Y., Geng, Z., Zhu, Q., and Han, Y. (2017). Review: Multi-objective optimization methods and application in energy saving. *Energy*, *125*, 681-704. https://doi.org/10.1016/J.ENERGY.2017.02.174

Çakır, Ö. (2008). *Veri madenciliğinde sınıflandırma yöntemlerinin karşılaştırılması "bankacılık müşteri veri tabanı üzerinde bir uygulama"* (Doktora). Marmara Üniversitesi, Sosyal Bilimler Enstitüsü, İstanbul.

Çelik, A. (2022). *Özellik seçimi için güncel meta sezgisel optimizasyonları kullanılarak yeni ikili optimizasyon algoritması geliştirilmesi* (Yüksek Lisans). Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elazığ.

Dash, M., and Liu, H. (2003). Consistency-based search in feature selection. *Artificial Intelligence*, *151*, 155-176. https://doi.org/10.1016/S0004-3702(03)00079-1

Dayan, P. (1999). Unsupervised learning. In R. A. Wilson and F. Keil (Ed.), *The MIT encyclopedia of the cognitive sciences* (1-7). The MIT Press, New York.

Deb, K. (2014). Multi-objective Optimization. In E. K. Burke and G. Kendall (Ed.), *Search Methodologies* (403-450). Springer, Boston.

Deb, K., and Goel, T. (1993). Controlled Elitist Non-dominated Sorting Genetic Algorithms for Better Convergence. In E. Zitzler, L. Thiele, K. Deb, C. A. Coello, and D. Corne (Ed.), *Evolutionary Multi-Criterion Optimization* (67-81). Springer, Berlin.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, *6*(2), 182-197. https://doi.org/10.1109/4235.996017

Dickey, G., Blanke, S., and Seaton, L. (2019). Machine learning in auditing. *The CPA Journal*, *89*(6), 16-21.

Dong, H., Sun, J., and Sun, X. (2021). A multi-objective multi-label feature selection algorithm based on shapley value. *Entropy*, *23*(8), 1094.

Dranca, L., de Abetxuko Ruiz de Mendarozketa, L., Goñi, A., Illarramendi, A., Navalpotro Gomez, I., Delgado Alvarado, M., and Cruz Rodríguez-Oroz, M. (2018). Using Kinect to classify Parkinson's disease stages related to severity of gait impairment. *BMC Bioinformatics*, *19*(1), 1-15. https://doi.org/10.1186/S12859-018-2488-4/TABLES/11

Elhariri, E., El-Bendary, N., and Taie, S. A. (2020). Using Hybrid Filter-Wrapper Feature Selection With Multi-Objective Improved-Salp Optimization for Crack Severity Recognition. *IEEE ACCESS*, *8*, 84290-84315. https://doi.org/10.1109/ACCESS.2020.2991968

El-kenawy, E.-S. M., Ibrahim, A., Mirjalili Seyedali and Eid, M. M., and Hussein, S. E. (2020). Novel Feature Selection and Voting Classifier Algorithms for COVID-19 Classification in CT Images. *IEEE ACCESS*, *8*, 179317-179335. https://doi.org/10.1109/ACCESS.2020.3028012

El-Kenawy, E.-S. M., Mirjalili, S., Ibrahim Abdelhameed and Alrahmawy, M., El-Said, M., Zaki, R. M., and Eid, M. M. (2021). Advanced Meta-Heuristics, Convolutional Neural Networks, and Feature Selectors for Efficient COVID-19 X-Ray Chest Image Classification. *IEEE ACCESS*, *9*, 36019-36037. https://doi.org/10.1109/ACCESS.2021.3061058

Emre, İ. E., Taş, C., ve Erol, Ç. (2021). Psikiyatride Makine Öğrenmesi Yöntemlerinin Kullanımı. *Psikiyatride Güncel Yaklaşımlar*, *13*(2), 332-353.

Ezirmik, A. H. (2020). *Meta-sezgisel yöntemler ile müzik verisi üzerinde özellik seçimi ve kategorizasyon* (Yüksek Lisans). Eskişehir Osmangazi Üniversitesi, Fen Bilimleri Enstitüsü, Eskişehir.

Fayyad, U. M., and Irani, K. B. (1993). Multi-lnterval Discretization of Continuous-Valued Attributes for Classification Learning. *Proceedings of 13th International Joint Conference on Artificial Intelligence*, Chambèry, 28 August-3 September 1993, 1022-1027.

Feng, Z., Huang, J., Jin, S., Wang, G., and Chen, Y. (2022). Artificial intelligence-based multi-objective optimisation for proton exchange membrane fuel cell: A literature review. *Journal of Power Sources*, *520*, 1-16. https://doi.org/10.1016/J.JPOWSOUR.2021.230808

Flach, P. (2012). *Machine Learning: The Art and Science of Algorithms That Make Sense of Data*. Cambridge University Press.

Frank, E., and Hall, M. A. (2011). *Data mining: practical machine learning tools and techniques*, Morgan Kaufmann. https://doi.org/10.1016/C2009-0-19715-5

Frank, E., Hall, M. A., and Witten, I. H. (2016). *The WEKA Workbench*, 4th ed., Morgan Kaufmann.

Freund, Y., and Schapire, R. E. (1996). Experiments with a new boosting algorithm. *13th International Conference on Machine Learning*, 28 June – 1 July 1996, Garda, Italy, 148-156.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of Statistics*, *29*(5), 1189-1232.

Geren, H. (2022). *A hybrid swarm intelligence algorithm for simultaneous feature selection and clustering* (Yüksek Lisans). Orta Doğu Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.

Geurts, P., Ernst, D., and Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning*, *63*, 3-42.

Ghaleb, S. A. A., Mohamad, M., Ghanem Waheed Ali H. M. and Nasser, A. B., Ghetas, M., Abdullahi Akibu Mahmoud and Saleh, S. A. M., Arshad, H., … Abiodun, O. I. (2022). Feature Selection by Multiobjective Optimization: Application to Spam Detection System by Neural Networks and Grasshopper Optimization Algorithm. *IEEE ACCESS*, *10*, 98475-98489. https://doi.org/10.1109/ACCESS.2022.3204593

Ghanem, W. A. H. M., Ghaleb, S. A. A., Jantan, A., Nasser, A. B., Saleh, S. A. M., Ngah, A., … Abiodun, O. I. (2022). Cyber Intrusion Detection System Based on a Multiobjective Binary Bat Algorithm for Feature Selection and Enhanced Bat Algorithm for Parameter Optimization in Neural Networks. *IEEE ACCESS*, *10*, 76318-76339. https://doi.org/10.1109/ACCESS.2022.3192472

Gill, P. E., Murray, W., and Wright, M. H. (1981). *Practical optimization*. Academic Press, California, USA.

Gunning, D., Stefik, M., Choi, J., Miller, T., Stumpf, S., and Yang, G. Z. (2019). XAI—Explainable artificial intelligence. *Science robotics*, *4*(37), eaay7120.

Guyon, I., Weston, J., and Barnhill, S. (2002). Gene Selection for Cancer Classification using Support Vector Machines. *Machine Learning*, *46*, 389-422.

Hall, M. A., ve Holmes, G. (2003). Benchmarking attribute selection techniques for discrete class data mining. *IEEE Transactions on Knowledge and Data engineering*, *15*(6), 1437-1447.

Hall, M.A. (2000). Correlation-based feature selection of discrete and numeric class machine learning. *Computer Science Working Papers*. New Zealand: University of Waikato. https://hdl.handle.net/10289/1024 [Ziyaret tarihi: 12.01.2022].

Hall, Mark A. (1999). *Correlation-based Feature Selection for Machine Learning* (Master, The University of Waikato). The University of Waikato, PhD. Thesis, Hamilton, NewZealand. https://www.cs.waikato.ac.nz/~mhall/thesis.pdf [Ziyaret tarihi: 29.05.2022].

Han, J., Kamber, M., and Pei, J. (2012). *Data mining concepts and techniques*, 3rd ed., University of Illinois at Urbana-Champaign Micheline Kamber Jian Pei Simon Fraser University.

Hart, P. E., Stork, D. G., and Duda, R. O. (2000). *Pattern classification*. Hoboken: Wiley.

Hastie, T., Tibshirani, R., Friedman, J. H., and Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction*, 2nd ed., Springer, New York. https://doi.org/10.1007/978-0-387-84858-7

Ho, T. K. (1995). Random decision forests. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, *1*, 278-282. IEEE Computer Society. https://doi.org/10.1109/ICDAR.1995.598994

Holland, J. H. (1992). Genetic Algorithms. *Scientific American*, 267(1), 66-73.

Hu, L., Li, Y., Gao, W., Zhang, P., and Hu, J. (2020). Multi-label feature selection with shared common mode. *Pattern Recognition*, *104*, 1-12. https://doi.org/10.1016/j.patcog.2020.107344

Jacobs, L. W., and Brusco, M. J. (1995). Note: A local-search heuristic for large set-covering problems. *Naval Research Logistics (NRL)*, *42*(7), 1129-1140.

Jiang, S. Y., ve Wang, L. X. (2016). Efficient feature selection based on correlation measure between continuous and discrete features. *Information Processing Letters*, *116*(2), 203-215.

Jiarpakdee, J., Tantithamthavorn, C., and Treude, C. (2020). The impact of automated feature selection techniques on the interpretation of defect models. *Empirical Software Engineering*, *25*, 3590-3638. https://doi.org/10.1007/s10664-020-09848-1

Jimenez, F., Sanchez, G., Palma, J., Miralles-Pechuan, L., and Botia, J. A. (2022). Multivariate Feature Ranking With High-Dimensional Data for Classification Tasks. *IEEE Access*, *10*, 60421-60437. https://doi.org/10.1109/ACCESS.2022.3180773

Kalaycı, G. (2019). *Application of metaheuristics for the feature selection problem* (Yüksek Lisans). Marmara Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.

Kalousis, A., Prados, J., and Hilario, M. (2007). Stability of feature selection algorithms: A study on high-dimensional spaces. *Knowledge and Information Systems*, *12*(1), 95-116. https://doi.org/10.1007/S10115-006-0040-8/METRICS

Kartal, E. (2021). Makine öğrenmesi ve yapay zeka. Içinde N. Bozbuğa ve S. Gülseçen (Ed.), *Tıp Bilişimi* (351-365). İstanbul University Press, İstanbul.

Kaya, S., ve Fiğlali, N. (2016). Çok Amaçlı Optimizasyon Problemlerinde Pareto Optimal Kullanımı. *Social Sciences Research Journal*, *5*(2), 9-18.

Kaygusuz, Y. (2021). *Türkçe metinlerin sınıflandırılmasında kullanılmak için geliştirilen melez bir öz nitelik seçim yöntemi* (Yüksek Lisans). Manisa Celal Bayar Üniversitesi, Fen Bilimleri Enstitüsü, Manisa.

Kennedy, J., and Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of International Conference on Neural Networks,* 27 Nov-1 Dec 1995, Perth, WA, Australia, 1942-1948.

Khalafallah, S. (2021). *Sınıflandırma algoritmalarında öznitelik seçimi için evrimsel algoritmaların kullanımı* (Yüksek Lisans). Ankara Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.

Khishe, M., & Mosavi, M. R. (2020). Chimp optimization algorithm. *Expert systems with applications*, *149*, 113338.

Kohavi, R., and John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, *97*, 273-324.

Kukkonen, S., and Deb, K. (2006). Improved pruning of non-dominated solutions based on crowding distance for bi-objective optimization problems. *2006 IEEE Congress on Evolutionary Computation, CEC 2006*, 1179-1186. https://doi.org/10.1109/cec.2006.1688443

Kurgan, L. A., Cios, K. J., Tadeusiewicz, R., Ogiela, M., and Goodenday, L. S. (2001). Knowledge discovery approach to automated cardiac SPECT diagnosis. *Artificial Intelligence in Medicine*, *23*(2), 149-19.

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., … Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*, *42*, 60-88.

Liu, H., and Motoda, H. (1998a). Feature Extraction, Construction and Selection: A Data Mining Perspective, *Feature Extraction, Construction and Selection*. Springer, New York. https://doi.org/10.1007/978-1-4615-5725-8

Liu, H., and Motoda, H. (1998b). Feature Selection for Knowledge Discovery and Data Mining. In H. Liu and H. Motoda (Ed.), *Feature Selection for Knowledge Discovery and Data Mining*. Springer, New York. https://doi.org/10.1007/978-1-4615-5689-3

Liu, H., and Setiono, R. (1996). A probabilistic approach to feature selection-a filter solution. *13th International Conference on Machine Learning*, *13th International Conference on Machine Learning*, 28 June – 1 July 1996, Garda, Italy, 319-327.

Liu, Z.-Z., Huang, J.-W., Wang, Y., and Cao, D.-S. (2018). ECoFFeS: A Software Using Evolutionary Computation for Feature Selection in Drug Discovery. *IEEE ACCESS*, *6*, 20950-20963. https://doi.org/10.1109/ACCESS.2018.2821441

Lu, J., Zhao, P., and Hoi, S. C. H. (2016). Online Passive-Aggressive Active learning. *Machine Learning*, *103*(2), 141-183. https://doi.org/10.1007/S10994-016-5555-Y/FIGURES/7

MacKay, D. (1956). The epistemological problem for automata. In C. Shannon ve J. McCarthy (Ed.), *Automata Studies* (235-251). Princeton University Press. https://doi.org/10.1515/9781400882618-012

McCulloch, W. S. , and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, *5*(4), 115-133.

Michalski, R. S., and Chilausky, R. L. (1980). Knowledge acquisition by encoding expert rules versus computer induction from examples: a case study involving soybean pathology. *International Journal of Man-Machine Studies*, *12*(1), 63-87.

Mirjalili, S., and Dong, J. S. (2020). *Multi-Objective Optimization using Artificial Intelligence Techniques*. Springer, Warsaw.

Mitchell, T. (1997). *Machine Learning*. McGraw-Hill, New York.

Mohri, M., Rostamizadeh, A., ve Talwalkar, A. (2018). *Foundations of Machine Learning*, 2nd ed., The MIT Press. ISBN: 978-0262039406.

Molina, L. C., Belanche, L., and Nebot, À. (2002). Feature Selection Algorithms: A Survey and Experimental Evaluation. *IEEE International Conference on Data Mining*, 306-313.

Ni, C., Chen, X., Wu, F., Shen, Y., and Gu, Q. (2019). An empirical study on pareto based multi-objective feature selection for software defect prediction. *Journal of Systems and Software*, *152*, 215-238. https://doi.org/10.1016/j.jss.2019.03.012

Noordewier, M., Towell, G., and Shavlik, J. (1990). Training knowledge-based neural networks to recognize genes in DNA sequences. *Advances in neural information processing systems*, *3*.

Othman, M. S., Kumaran, S. R., and Yusuf, L. M. (2020). Gene Selection Using Hybrid Multi-Objective Cuckoo Search Algorithm With Evolutionary Operators for Cancer Microarray Data. *IEEE ACCESS*, *8*, 186348-186361. https://doi.org/10.1109/ACCESS.2020.3029890

Özkan, Y., ve Erol, Ç. (2015). *Biyoenformatik DNA Mikrodizi Veri Madenciliği*. Papatya yayınevi, İstanbul.

Özmen, M. (2016). *Interactive evolutionary approaches to multi-objective feature selection* (Yüksek Lisans). Orta Doğu Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.

Pedrycz, W., Succi, G., and Sillitti, A. (2016). *Computational Intelligence and quantitative software engineering*. Springer, Cham. https://doi.org/10.1007/978-3-319-25964-2

Piri, J., Mohapatra, P., Singh, H. K. R., Acharya, B., and Patra, T. K. (2022a). An Enhanced Binary Multiobjective Hybrid Filter-Wrapper Chimp Optimization Based Feature Selection Method for COVID-19 Patient Health Prediction. *IEEE ACCESS*, *10*, 100376-100396. https://doi.org/10.1109/ACCESS.2022.3203400

Piri, J., Mohapatra, P., Singh, D., Samanta, D., Singh, D., Kaur, M., and Lee, H.-N. (2022b). Mining and Interpretation of Critical Aspects of Infant Health Status Using Multi-Objective Evolutionary Feature Selection Approaches. *IEEE ACCESS*, *10*, 32622-32638. https://doi.org/10.1109/ACCESS.2022.3161154

Piri, J., Mohapatra, P., Pradhan, M. R., Acharya, B., and Patra, T. K. (2022c). A Binary Multi-Objective Chimp Optimizer With Dual Archive for Feature Selection in the Healthcare Domain. *IEEE ACCESS*, *10*, 1756-1774. https://doi.org/10.1109/ACCESS.2021.3138403

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., and Gulin, A. (2018). CatBoost: unbiased boosting with categorical features. *Advances in neural information processing systems,* 31, 1-11.

Quinlan, J. R. (1987). *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Francisco.

Raj, R. J. S., Shobana, S. J., Pustokhina, I. V., Pustokhin, D. A., Gupta, D., and Shankar, K. (2020). Optimal Feature Selection-Based Medical Image Classification Using Deep Learning Model in Internet of Medical Things. *IEEE ACCESS*, *8*, 58006-58017. https://doi.org/10.1109/ACCESS.2020.2981337

Rodrigues, D., Pereira, L. A., Almeida, T. N. S., Papa, J. P., Souza, A. N., Ramos, C. C., and Yang, X. S. (2013, May). BCS: A binary cuckoo search algorithm for feature selection. In *2013 IEEE International symposium on circuits and systems (ISCAS)*, 19-23 May 2013, Beijing, China, 465-468.

Salimi, H. (2015). Stochastic Fractal Search: A powerful metaheuristic algorithm. *Knowledge-Based Systems*, *75*, 1-18.

Schlimmer, J. C. (1987). *Concept acquisition through representational adjustment*. University of California, Irvine.

Sezer, E. (2018). *Sınıflandırma sorunu için en uygun değişken alt kümesi seçimi üzerine bir uygulama* (Yüksek Lisans). Marmara Üniversitesi, Sosyal Bilimler Enstitüsü, İstanbul.

Sezer, E., and Başeğmez, H. (2022). An Approach Based on Feature Selection for Missing Value Imputation. In S. and C. O. S. and O. B. and T. A. C. and S. I. U. Kahraman Cengiz and Cebi (Ed.), *Intelligent and Fuzzy Techniques for Emerging Conditions and Digital Transformation* (945-950). Springer International Publishing, Cham. https://doi.org/10.1007/978-3-030-85626-7_110

Sezer, E., and Çakır, Ö. (2022). A Feature Selection Application for Classification: A Banking Application. *Dicle University Journal of Economics and Administrative Sciences*, *12*(24), 480-498.

Shapiro, A. D. (1984). *The role of structured induction in expert systems*. Annexe Thesis Digitisation Project, The University of Edinburgh.

Srinivas, N., and Deb, K. (1994). Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. *Evolutionary Computation*, *2*(3), 221-248. https://doi.org/10.1162/EVCO.1994.2.3.221

Subaş, N. (2019). *Sürekli/ikili parçacık sürü optimizasyonu ve destek vektör makinelerinin hibrit kullanımı ile özellik seçimi* (Yüksek Lisans). Mimar Sinan Güzel Sanatlar Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.

Sugiyama, M. (2015). *Statistical Reinforcement Learning: Modern Machine Learning Approaches*. CRC Press, Taylor & Francis Group, Florida.

Sun, Y., Xiong, W., Yao, Z., Moniz, K., & Zahir, A. (2018). Analysis of network attack and defense strategies based on Pareto optimum. Electronics, 7(3), 36.

Sutton, R. S., and Barto, A. G. (2018). *Reinforcement Learning: An Introduction*, 2nd ed., The MIT Press, London.

Thiele, L., Miettinen, K., Korhonen, P. J., and Molina, J. (2009). A preference-based evolutionary algorithm for multi-objective optimization. *Evolutionary computation*, *17*(3), 411-436.

Usman, A. M., Yusof, U. K., and Naim, S. (2020). Filter-Based Multi-Objective Feature Selection Using NSGA III and Cuckoo Optimization Algorithm. *IEEE ACCESS*, *8*, 76333-76356. https://doi.org/10.1109/ACCESS.2020.2987057

Witten, I. H., Frank, E., ve Hall, M. A. (2011). *Data Mining: Practical Machine Learning Tools and Techniques* (C. 2). Morgan Kaufmann, San Francisco.

Xu, Y., Wang, J., An, S., Wei, J., and Ruan, J. (2018). Semi-supervised multi-label feature selection by preserving feature-label space consistency. *International Conference on Information and Knowledge Management, Proceedings*, 783-792. Association for Computing Machinery. https://doi.org/10.1145/3269206.3271760

Yu, L., and Liu, H. (2003). Feature selection for high-dimensional data: A fast correlation-based filter solution. *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, 856-863.

Zhang, Y., Zhang, Y., Zhang, C., and Zhou, C. (2022). Multiobjective Harris Hawks Optimization With Associative Learning and Chaotic Local Search for Feature Selection. *IEEE ACCESS*, *10*, 72973-72987. https://doi.org/10.1109/ACCESS.2022.318947

Zhang, Z. F., Liu, W., Ostrosi, E., Tian, Y., ve Yi, J. (2021). Steel strip surface inspection through the combination of feature selection and multiclass classifiers. *Engineering Computations*, *38*(4), 1831-1850. https://doi.org/10.1108/EC-11-2019-0502

# EKLER

## EK-1. ÇOKOTUÖS ALGORİTMASI

```
#1 Verilen bir kümenin n elemanlı alt kümelerinin bulunması
Tanımlama Altküme(s, n):
Girdi: s küme, n alt küme eleman sayısı
Çıktı: Altkümeler
  Altkümeler = kombinasyon(s, n)

#2 Tüm alt kümelerin bulunması
Tanımlama : TümAltkümeler
Girdi : Küme
Çıktı : tüm_altkümeler
    tüm_altkümeler = [] # Bir boş liste tanımlanır
    # Eleman sayısı bir olanlardan başlayarak kümenin eleman sayısından
bir eksik sayıda olanlara kadar tüm altkümelerin bulunması
    for i in range(elemansayısı(Küme)-1):
        altkümeler = Altkümeler(Küme, i+1)
        for j in altkümeler:
          tüm_altkümeler.ekle(j)

#3 Bilgi kazanımı
Tanımlama : Bilgi_Kazanımı

Girdi : f1,f2
Çıktı : bk

başla
  bk = entropi(f1) - koşullu_entropi(f1, f2)
bitir

#4 Simetrik Belirsizlik
Tanımlama : Simetrik_Belirsizlik

Girdi : f1,f2
Çıktı : sb

başla
    t1 = Bilgi_Kazanımı(f1, f2)
    t2 = entropi(f1)
    t3 = entropi(f2)

    sb = 2.0 * t1 / (t2 + t3)
bitir

#5 CFS merit değeri hesaplama
```

```
Tanımlama : CFS

Girdi: X  öznitelik kümesi, y  Sınıf değerleri
Çıktı: merit_cfs

başla
  rff = 0 öznitelikler arası korelasyon toplamı için başlangıç değeri
sıfır olsun
  rcf = 0 öznitelikler ile sınıf arasındaki korelasyon için başlangıç
değeri sıfır olsun

  sayaç = 0
  her f eleman X için:
    her g eleman X için:
      f eşit değil g olduğunda rff = rff + Simetrik_Belirsizlik(f,g)
      sayaç = sayaç + 1

  rff = rff / sayaç # öznitelikler arası korelasyon ortalaması

  sayaç = 0
  her f eleman X için:
    rcf = (rcf + Simetrik_Belirsizlik(f,y))
    sayaç = sayaç + 1

  rcf = rcf / sayaç # öznitelikler ile sınıf korelasyon ortalaması

  k : öznitelik sayısı olmak üzere

  merit_cfs = (k * rcf) / karekök(k + k * (k-1) * rff)

bitir

#6 Verilen veri kümesi ve sınıf ile tutarlılık hesaplama fonksiyonu
Tanımlama : CONS

Girdi
  Veri : (f1,f2, ... fn) şeklinde n değişkenden oluşan veri kümesi,
  Sınıf: (s1,s2,...sn) şeklinde benzersiz değer alan veri kümesi
içerisindeki sınıf değişkeni

Çıktı
  merit_tutarlılık : tutarlılık merit değeri

Başla

  Veri1 = Veri - Sınıf # Veri içerisindeki sınıf dışında kalan
öznitelik kümesi belirlenir
  Veri1 içerisindeki ilk satırdaki değerler (a1,a2,...,an-1) olsun
```

```
  ilk satır ile aynı değerleri alan satır sayısı m olsun ve
  bu satırlara karşılık gelen sınıf etiketlerinin tekrarlanma sayısı
(t1,t2,...tn-1) olsun.
  bu durumda t1 + t2 + .... + tn-1 = m eşitliği sağlanır
  r = maksimum(t1, t2,...,tn-1) olsun
  tutarsızlık = m - r ile bu benzersiz satırlar için tutarsızlık adedi
elde edilir
  tutarsızlık oranı = ((m - r) / m)

  Veri1 içerisinde benzersiz (f1,f2,..,fn) değer sayısı i olsun
  1 den i ye kadar her benzersiz (f1,f2,...,fn) için tutarsızlık
oranlarının ortalaması tutarsızlık_veri1 olsun
  tutarsızlık_veri1 Veri1in tutarsızlık değeridir

  merit_tutarlılık = (1 - tutarsızlık_veri1) ile Verinin tutarlılığı
hesaplanır

Bitir
```

```
#7 NSGAII (Deb,2002)
```

NSGA II *(P)*

for each $p \in P$
$S_p = \emptyset$
$n_p = 0$
for each $q \in P$
  if ($p \prec$ q) then  #p q'ya baskındır
    $S_p = S_p \cup \{q\}$  # p'yi baskılayan çözümlerin kümesini q ya ekle
  else if ($q \prec$ p) then
    $n_p = n_p + 1$  # p nin baskınlık sayacını bir artır
if $n_p = 0$ then  #p ilk cepheye aittir
  $p_{rank} = 1$
$\mathcal{F}_1 = \mathcal{F}_1 \cup \{p\}$
$i = 1$  # ön sayacı başlat
while $\mathcal{F}_i \neq \emptyset$
$Q = \emptyset$  # sonraki cephenin üyelerini depolamak için kullanıldı
For each $q \in S_p$
$n_q = n_q - 1$
if $n_q = 0$ then  # q sonraki cepheye aittir
$q_{rank} = i + 1$
$Q = Q \cup \{q\}$
$i = i + 1$
$\mathcal{F}_i = Q$

Kalabalık mesafe sıralama ($\mathcal{I}$)
$l = |\mathcal{I}|$  # $\mathcal{I}$'daki çözümlerin sayısı
for each $i$, set $\mathcal{I}[i]_{distance} = 0$  #mesafeyi başlat
for each objective $m$
$\mathcal{I} = sort(\mathcal{I}, m)$  # her bir amaç değerini kullanarak sıralama
$\mathcal{I}[1]_{distance} = \mathcal{I}[l]_{distance} = \infty$  # daima sınır noktaları seçebilmek için
for $i = 2$ to $(l - 1)$  #diğer bütün noktalar için
$\mathcal{I}[i]_{distance} = \mathcal{I}[i]_{distance} + (\mathcal{I}[i+1].m - \mathcal{I}[i-1].m)/(f_m^{\max} - f_m^{\min})$

```
#8 pareto_set (Sun, 2018)
```

**Procedure** OPTIMIZE(null)
for $i = 0,\ i++,\ i < Def_{st}.length$ **do**
$d = Def_{st}[i]$
**for** $j = 0,\ j++,\ j < Att_{st}.length$ **do**
$a = Att_{st}[j]$
$addToFront = true$
**for** $k = 0,\ k++, k < Att_{st}.length$ **do**
$a' = Att_{st}[k]$
**if** $j! = k$ **then**
**if** $P[0][a][d] <= P[0][a'][d]\ and\ P[0][a'][d] <= P[0][a'][d]$ **then**
$addToFront = false$
**end if**
**end if**
**end for**
**if** $addToFront$ **then**
$add(Fronts, d)$
**end if**
**end for**
**end for**
**for** $i = 0,\ i++,\ i < Fronts_{def}.legth$ **do**
$\boldsymbol{d = Fronts_{def}[i]}$
**for** $j = 0, j++,\ j < Fronts_{def}.length$ **do**
$d' = Fronts_{def}[j]$
**if** $j! = k$ **then**
for $k = 0, k++, k < Fronts_{att}[d].length$ **do**
$a' = Fronts_{att}[d][k]$
$dominant = true$
for $l = 0,\ l++,\ l < Fronts_{att}[d'].length$ **do**
$a' = Fronts_{att}[d'][k]$
**if** $P[0][a][d] <= P[0][a'][d]$ **and** $P[0][a'][d] <= P[0][a'][d']$ then
$dominant = flase$
end if
end for
if $dominant$ then
$remove(Fronts, d)$
end if
end for
end if
end for
**end for**

```
#9 Sarmalayıcı öznitelik alt kümesi seçimi
Tanımlama : Sarmalayıcı
Girdi: KümeUzayı
```

```
Çıktı: Küme

başla
    acc = 0
    for i in KümeUzayı:
      test_set = random_sampling(i,0.3) # %30 test için ayrılır
      train_set = i - test_set # %70 öğrenme için ayrılır
      model = sınıflandırıcı(train_set)
      tahmin = model(test_set)
      acc1 = doğruluk(tahmin,test_set) # test kümesi üzerinde tahmin
doğruluğu hesaplanır
      if acc1 > acc :
          Küme = i
bitir

#10 Verilen herhangi bir veri kümesi için çokotuös_pareto ve
çokotuös_nsgaii
Girdi : Küme
Çıktı : çokotuös_pareto, çokotuös_nsgaii
başla
    #1 Sürekli öznitelikleri kesikleştirilmesi sağlanır
    for öznitelik in küme:
      if type(öznitelik) == sürekli:
          öznitelik = kesikleştir(öznitelik)
    #2 verilen kümenin tüm alt kümelerinin bulunması sağlanır
    tüm_altkümeler = TümAltkümeler(Küme)

    #3 Altküme uzayından rastgele n adet adayın seçilmesi sağlanır
    aday_altkümeler = rastgele(tüm_altkümeler, n)

    consistency_merit = [] # boş cons merit değeri listesi tanımlanır
    correlation_merit = [] # boş cfs merit değeri listesi tanımlanır

    #4 merit değerlerinin hesaplanması ve listelere kaydedilmesi
sağlanır
    for j in aday_altkümeler:
        correlation_merit.ekle(CFS(j))
        consistency_merit.ekle(CONS(j))

    #5 CFS ve CONS için aynı anda maksimize hedefli pareto kümenin
bulunması sağlanır
    pareto_set = paretoset(correlation_merit), consistency_merit,
hedefler = (max, max))

    #6 CFS ve CONS için aynı anda maksimize hedefli nsgaii kümenin
bulunması sağlanır
    nsgaii_set = nsgaII(correlation_merit, consistency_merit, hedefler
= (max, max))
```

```
    #7 Sarmalayıcı yaklaşımla en yüksek accuracy elde eden pareto_set
içerisindeki aday çokotuös_pareto tarafından seçilen öznitelik alt
kümesi olarak atanır
    çokotuös_pareto = SarmalayıcıÖS(pareto_set)

    #8 Sarmalayıcı yaklaşımla en yüksek accuracy elde eden nsgaii_set
içerisindeki aday Çokotuös_nsgaii tarafından seçilen öznitelik alt
kümesi olarak atanır
    çokotuös_nsgaii = SarmalayıcıÖS(nsgaii_set)
bitir
```

## EK-2. Veri Kümesine Göre En Yüksek Doğruluk Sağlayan Sınıflandırıcılar

| **Veri Kümesi** | **En Yüksek Doğruluk Sağlayan Sınıflandırıcı** |
|---|---|
| Audiology (Standardized) | ExtraTreesClassifier |
| Breast Cancer | RandomForestClassifier |
| Chess (King-Rook vs. King-Pawn) | RandomForestClassifier |
| Congressional Voting Records | CatBoost<br>RandomForestClassifier<br>SVC |
| Lymphography | LogisticRegression |
| Mushroom | RandomForestClassifier<br>CatBoost<br>LGBM |
| Primary Tumor | CatBoost |
| Soybean | CatBoost<br>RandomForestClassifier |
| SPECT Heart | CatBoost |
| Splice-junction Gene Sequences | LGBM |

## EK-3. DETAYLI SINIFLANDIRMA SONUÇLARI

Audiology Veri Kümesi Öznitelik Seçimi Sınıflandırma Sonuçları

| Model | Accuracy | F1 Score | Selector |
|---|---|---|---|
| RandomForestClassifier | 70.59 | 0.67 | Mococo_Pareto |
| GaussianNB | 69.12 | 0.69 | Mococo_Pareto |
| XGBClassifier | 70.59 | 0.69 | Mococo_Pareto |
| DecisionTreeClassifier | 77.94 | 0.75 | Mococo_Pareto |
| LGBMClassifier | 75 | 0.72 | Mococo_Pareto |
| PassiveAggressiveClassifier | 63.24 | 0.58 | Mococo_Pareto |
| LogisticRegression | 52.94 | 0.45 | Mococo_Pareto |
| KNeighborsClassifier | 48.53 | 0.39 | Mococo_Pareto |
| SVC | 38.24 | 0.26 | Mococo_Pareto |
| SGDClassifier | 57.35 | 0.49 | Mococo_Pareto |
| ExtraTreesClassifier | 79.41 | 0.78 | Mococo_Pareto |
| CatBoost | 73.53 | 0.74 | Mococo_Pareto |
| RandomForestClassifier | 80.88 | 0.77 | Mococo_Nsgaii |
| GaussianNB | 69.12 | 0.67 | Mococo_Nsgaii |
| XGBClassifier | 75 | 0.73 | Mococo_Nsgaii |
| DecisionTreeClassifier | 80.88 | 0.78 | Mococo_Nsgaii |
| LGBMClassifier | 66.18 | 0.65 | Mococo_Nsgaii |
| PassiveAggressiveClassifier | 63.24 | 0.59 | Mococo_Nsgaii |
| LogisticRegression | 61.76 | 0.57 | Mococo_Nsgaii |
| KNeighborsClassifier | 52.94 | 0.45 | Mococo_Nsgaii |
| SVC | 45.59 | 0.37 | Mococo_Nsgaii |
| SGDClassifier | 54.41 | 0.49 | Mococo_Nsgaii |
| ExtraTreesClassifier | 82.35 | 0.8 | Mococo_Nsgaii |
| CatBoost | 82.35 | 0.8 | Mococo_Nsgaii |
| RandomForestClassifier | 70.59 | 0.67 | CFS |
| GaussianNB | 69.12 | 0.69 | CFS |

| | | | |
|---|---|---|---|
| XGBClassifier | 70.59 | 0.69 | CFS |
| DecisionTreeClassifier | 77.94 | 0.75 | CFS |
| LGBMClassifier | 75 | 0.72 | CFS |
| PassiveAggressiveClassifier | 63.24 | 0.58 | CFS |
| LogisticRegression | 52.94 | 0.45 | CFS |
| KNeighborsClassifier | 48.53 | 0.39 | CFS |
| SVC | 38.24 | 0.26 | CFS |
| SGDClassifier | 57.35 | 0.49 | CFS |
| ExtraTreesClassifier | 79.41 | 0.78 | CFS |
| CatBoost | 73.53 | 0.74 | CFS |
| RandomForestClassifier | 82.35 | 0.79 | Consistency |
| GaussianNB | 70.59 | 0.7 | Consistency |
| XGBClassifier | 72.06 | 0.7 | Consistency |
| DecisionTreeClassifier | 73.53 | 0.72 | Consistency |
| LGBMClassifier | 69.12 | 0.66 | Consistency |
| PassiveAggressiveClassifier | 58.82 | 0.52 | Consistency |
| LogisticRegression | 51.47 | 0.43 | Consistency |
| KNeighborsClassifier | 51.47 | 0.43 | Consistency |
| SVC | 32.35 | 0.19 | Consistency |
| SGDClassifier | 51.47 | 0.42 | Consistency |
| ExtraTreesClassifier | 83.82 | 0.81 | Consistency |
| CatBoost | 23.53 | 0.23 | Consistency |

Audiology Veri Kümesi En Yüksek Sınıflandırma Doğruluğu Sağlayan Algoritmalar

| | Selector | Accuracy |
|---|---|---|
| **ExtraTreesClassifier** | Consistency | 83.82 |
| **ExtraTreesClassifier** | Mococo_Nsgaii | 82.35 |
| **ExtraTreesClassifier** | Mococo_Pareto | 79.41 |
| **ExtraTreesClassifier** | CFS | 79.41 |

Breast Veri Kümesi Öznitelik Seçimi Sınıflandırma Sonuçları

| Model | Accuracy | F1 Score | Selector |
|---|---|---|---|
| **RandomForestClassifier** | 74.42 | 0.72 | Mococo_Nsgaii |
| **GaussianNB** | 41.86 | 0.39 | Mococo_Nsgaii |
| **XGBClassifier** | 67.44 | 0.66 | Mococo_Nsgaii |
| **DecisionTreeClassifier** | 70.93 | 0.7 | Mococo_Nsgaii |
| **LGBMClassifier** | 69.77 | 0.68 | Mococo_Nsgaii |
| **PassiveAggressiveClassifier** | 59.3 | 0.56 | Mococo_Nsgaii |
| **LogisticRegression** | 68.6 | 0.66 | Mococo_Nsgaii |
| **KNeighborsClassifier** | 74.42 | 0.72 | Mococo_Nsgaii |
| **SVC** | 74.42 | 0.71 | Mococo_Nsgaii |
| **SGDClassifier** | 60.47 | 0.6 | Mococo_Nsgaii |
| **ExtraTreesClassifier** | 74.42 | 0.72 | Mococo_Nsgaii |
| **CatBoost** | 72.09 | 0.7 | Mococo_Nsgaii |
| **RandomForestClassifier** | 73.26 | 0.71 | Mococo_Pareto |
| **GaussianNB** | 73.26 | 0.72 | Mococo_Pareto |
| **XGBClassifier** | 72.09 | 0.7 | Mococo_Pareto |
| **DecisionTreeClassifier** | 65.12 | 0.65 | Mococo_Pareto |
| **LGBMClassifier** | 68.6 | 0.66 | Mococo_Pareto |
| **PassiveAggressiveClassifier** | 55.81 | 0.55 | Mococo_Pareto |
| **LogisticRegression** | 67.44 | 0.65 | Mococo_Pareto |
| **KNeighborsClassifier** | 63.95 | 0.62 | Mococo_Pareto |
| **SVC** | 70.93 | 0.68 | Mococo_Pareto |
| **SGDClassifier** | 61.63 | 0.61 | Mococo_Pareto |
| **ExtraTreesClassifier** | 72.09 | 0.7 | Mococo_Pareto |
| **CatBoost** | 73.26 | 0.71 | Mococo_Pareto |
| **RandomForestClassifier** | 70.93 | 0.69 | CFS |
| **GaussianNB** | 72.09 | 0.71 | CFS |
| **XGBClassifier** | 70.93 | 0.69 | CFS |
| **DecisionTreeClassifier** | 63.95 | 0.64 | CFS |

| | | | |
|---|---|---|---|
| **LGBMClassifier** | 70.93 | 0.69 | CFS |
| **PassiveAggressiveClassifier** | 55.81 | 0.55 | CFS |
| **LogisticRegression** | 67.44 | 0.65 | CFS |
| **KNeighborsClassifier** | 61.63 | 0.58 | CFS |
| **SVC** | 69.77 | 0.67 | CFS |
| **SGDClassifier** | 65.12 | 0.64 | CFS |
| **ExtraTreesClassifier** | 72.09 | 0.7 | CFS |
| **CatBoost** | 69.77 | 0.67 | CFS |
| **RandomForestClassifier** | 72.09 | 0.7 | Consistency |
| **GaussianNB** | 40.7 | 0.35 | Consistency |
| **XGBClassifier** | 69.77 | 0.68 | Consistency |
| **DecisionTreeClassifier** | 68.6 | 0.68 | Consistency |
| **LGBMClassifier** | 69.77 | 0.68 | Consistency |
| **PassiveAggressiveClassifier** | 60.47 | 0.6 | Consistency |
| **LogisticRegression** | 69.77 | 0.68 | Consistency |
| **KNeighborsClassifier** | 70.93 | 0.68 | Consistency |
| **SVC** | 73.26 | 0.7 | Consistency |
| **SGDClassifier** | 69.77 | 0.69 | Consistency |
| **ExtraTreesClassifier** | 73.26 | 0.71 | Consistency |
| **CatBoost** | 72.09 | 0.7 | Consistency |

Breast Veri Kümesi En Yüksek Sınıflandırma Doğruluğu Sağlayan Algoritmalar

| | Selector | Accuracy |
|---|---|---|
| **RandomForestClassifier** | Mococo_Nsgaii | 74.42 |
| **RandomForestClassifier** | Mococo_Pareto | 73.26 |
| **SVC** | Consistency | 73.26 |
| **GaussianNB** | CFS | 72.09 |

Chess Veri Kümesi Öznitelik Seçimi Sınıflandırma Sonuçları

| Model | Accuracy | F1 Score | Selector |
|---|---|---|---|

| | | | |
|---|---|---|---|
| RandomForestClassifier | 99.27 | 0.99 | Mococo_Pareto |
| GaussianNB | 60.38 | 0.53 | Mococo_Pareto |
| XGBClassifier | 97.6 | 0.98 | Mococo_Pareto |
| DecisionTreeClassifier | 99.17 | 0.99 | Mococo_Pareto |
| LGBMClassifier | 99.06 | 0.99 | Mococo_Pareto |
| PassiveAggressiveClassifier | 94.16 | 0.94 | Mococo_Pareto |
| LogisticRegression | 96.98 | 0.97 | Mococo_Pareto |
| KNeighborsClassifier | 94.16 | 0.94 | Mococo_Pareto |
| SVC | 98.02 | 0.98 | Mococo_Pareto |
| SGDClassifier | 96.66 | 0.97 | Mococo_Pareto |
| ExtraTreesClassifier | 98.96 | 0.99 | Mococo_Pareto |
| CatBoost | 98.96 | 0.99 | Mococo_Pareto |
| RandomForestClassifier | 99.06 | 0.99 | Mococo_Nsgaii |
| GaussianNB | 61 | 0.54 | Mococo_Nsgaii |
| XGBClassifier | 97.6 | 0.98 | Mococo_Nsgaii |
| DecisionTreeClassifier | 99.06 | 0.99 | Mococo_Nsgaii |
| LGBMClassifier | 99.06 | 0.99 | Mococo_Nsgaii |
| PassiveAggressiveClassifier | 94.16 | 0.94 | Mococo_Nsgaii |
| LogisticRegression | 96.98 | 0.97 | Mococo_Nsgaii |
| KNeighborsClassifier | 94.16 | 0.94 | Mococo_Nsgaii |
| SVC | 98.12 | 0.98 | Mococo_Nsgaii |
| SGDClassifier | 96.56 | 0.97 | Mococo_Nsgaii |
| ExtraTreesClassifier | 98.96 | 0.99 | Mococo_Nsgaii |
| CatBoost | 98.33 | 0.98 | Mococo_Nsgaii |
| RandomForestClassifier | 99.06 | 0.99 | CFS |
| GaussianNB | 63.19 | 0.58 | CFS |
| XGBClassifier | 97.6 | 0.98 | CFS |
| DecisionTreeClassifier | 99.06 | 0.99 | CFS |
| LGBMClassifier | 98.96 | 0.99 | CFS |

| | | | |
|---|---|---|---|
| PassiveAggressiveClassifier | 93.95 | 0.94 | CFS |
| LogisticRegression | 96.98 | 0.97 | CFS |
| KNeighborsClassifier | 93.33 | 0.93 | CFS |
| SVC | 97.6 | 0.98 | CFS |
| SGDClassifier | 96.77 | 0.97 | CFS |
| ExtraTreesClassifier | 98.75 | 0.99 | CFS |
| CatBoost | 71.53 | 0.69 | CFS |
| RandomForestClassifier | 99.27 | 0.99 | Consistency |
| GaussianNB | 61.31 | 0.54 | Consistency |
| XGBClassifier | 97.6 | 0.98 | Consistency |
| DecisionTreeClassifier | 99.17 | 0.99 | Consistency |
| LGBMClassifier | 99.06 | 0.99 | Consistency |
| PassiveAggressiveClassifier | 94.47 | 0.94 | Consistency |
| LogisticRegression | 96.98 | 0.97 | Consistency |
| KNeighborsClassifier | 93.33 | 0.93 | Consistency |
| SVC | 98.02 | 0.98 | Consistency |
| SGDClassifier | 96.14 | 0.96 | Consistency |
| ExtraTreesClassifier | 99.06 | 0.99 | Consistency |
| CatBoost | 98.75 | 0.99 | Consistency |

Chess Veri Kümesi En Yüksek Sınıflandırma Doğruluğu Sağlayan Algoritmalar

| | Selector | Accuracy |
|---|---|---|
| **RandomForestClassifier** | Mococo_Pareto | 99.27 |
| **RandomForestClassifier** | Consistency | 99.27 |
| **RandomForestClassifier** | Mococo_Nsgaii | 99.06 |
| **RandomForestClassifier** | CFS | 99.06 |

Lymphography Veri Kümesi Öznitelik Seçimi Sınıflandırma Sonuçları

| Model | Accuracy | F1 Score | Selector |
|---|---|---|---|
| RandomForestClassifier | 81.82 | 0.81 | Mococo_Pareto |

| | | | |
|---|---|---|---|
| GaussianNB | 75 | 0.74 | Mococo_Pareto |
| XGBClassifier | 75 | 0.74 | Mococo_Pareto |
| DecisionTreeClassifier | 72.73 | 0.74 | Mococo_Pareto |
| LGBMClassifier | 86.36 | 0.84 | Mococo_Pareto |
| PassiveAggressiveClassifier | 84.09 | 0.85 | Mococo_Pareto |
| LogisticRegression | 88.64 | 0.89 | Mococo_Pareto |
| KNeighborsClassifier | 88.64 | 0.89 | Mococo_Pareto |
| SVC | 79.55 | 0.78 | Mococo_Pareto |
| SGDClassifier | 81.82 | 0.82 | Mococo_Pareto |
| ExtraTreesClassifier | 77.27 | 0.76 | Mococo_Pareto |
| CatBoost | 79.55 | 0.78 | Mococo_Pareto |
| RandomForestClassifier | 81.82 | 0.8 | Mococo_Nsgaii |
| GaussianNB | 72.73 | 0.72 | Mococo_Nsgaii |
| XGBClassifier | 81.82 | 0.8 | Mococo_Nsgaii |
| DecisionTreeClassifier | 81.82 | 0.81 | Mococo_Nsgaii |
| LGBMClassifier | 84.09 | 0.82 | Mococo_Nsgaii |
| PassiveAggressiveClassifier | 79.55 | 0.78 | Mococo_Nsgaii |
| LogisticRegression | 77.27 | 0.77 | Mococo_Nsgaii |
| KNeighborsClassifier | 88.64 | 0.89 | Mococo_Nsgaii |
| SVC | 84.09 | 0.82 | Mococo_Nsgaii |
| SGDClassifier | 81.82 | 0.82 | Mococo_Nsgaii |
| ExtraTreesClassifier | 81.82 | 0.8 | Mococo_Nsgaii |
| CatBoost | 81.82 | 0.82 | Mococo_Nsgaii |
| RandomForestClassifier | 81.82 | 0.81 | CFS |
| GaussianNB | 75 | 0.74 | CFS |
| XGBClassifier | 75 | 0.74 | CFS |
| DecisionTreeClassifier | 72.73 | 0.74 | CFS |
| LGBMClassifier | 86.36 | 0.84 | CFS |
| PassiveAggressiveClassifier | 84.09 | 0.85 | CFS |

| | | | |
|---|---|---|---|
| LogisticRegression | 88.64 | 0.89 | CFS |
| KNeighborsClassifier | 88.64 | 0.89 | CFS |
| SVC | 79.55 | 0.78 | CFS |
| SGDClassifier | 81.82 | 0.82 | CFS |
| ExtraTreesClassifier | 77.27 | 0.76 | CFS |
| CatBoost | 79.55 | 0.78 | CFS |
| RandomForestClassifier | 84.09 | 0.82 | Consistency |
| GaussianNB | 72.73 | 0.72 | Consistency |
| XGBClassifier | 79.55 | 0.78 | Consistency |
| DecisionTreeClassifier | 77.27 | 0.79 | Consistency |
| LGBMClassifier | 84.09 | 0.82 | Consistency |
| PassiveAggressiveClassifier | 88.64 | 0.89 | Consistency |
| LogisticRegression | 93.18 | 0.93 | Consistency |
| KNeighborsClassifier | 86.36 | 0.86 | Consistency |
| SVC | 81.82 | 0.8 | Consistency |
| SGDClassifier | 77.27 | 0.78 | Consistency |
| ExtraTreesClassifier | 75 | 0.74 | Consistency |
| CatBoost | 72.73 | 0.71 | Consistency |

Lymphography Veri Kümesi En Yüksek Sınıflandırma Doğruluğu Sağlayan Algoritmalar

| | Selector | Accuracy |
|---|---|---|
| **LogisticRegression** | Consistency | 93.18 |
| **LogisticRegression** | Mococo_Pareto | 88.64 |
| **KNeighborsClassifier** | Mococo_Nsgaii | 88.64 |
| **LogisticRegression** | CFS | 88.64 |

Mushroom Veri Kümesi Öznitelik Seçimi Sınıflandırma Sonuçları

| Model | Accuracy | F1 Score | Selector |
|---|---|---|---|
| RandomForestClassifier | 100 | 1 | Mococo_Pareto |

| | | | |
|---|---|---|---|
| GaussianNB | 99.75 | 1 | Mococo_Pareto |
| XGBClassifier | 100 | 1 | Mococo_Pareto |
| DecisionTreeClassifier | 100 | 1 | Mococo_Pareto |
| LGBMClassifier | 100 | 1 | Mococo_Pareto |
| PassiveAggressiveClassifier | 100 | 1 | Mococo_Pareto |
| LogisticRegression | 100 | 1 | Mococo_Pareto |
| KNeighborsClassifier | 100 | 1 | Mococo_Pareto |
| SVC | 100 | 1 | Mococo_Pareto |
| SGDClassifier | 100 | 1 | Mococo_Pareto |
| ExtraTreesClassifier | 100 | 1 | Mococo_Pareto |
| CatBoost | 100 | 1 | Mococo_Pareto |
| RandomForestClassifier | 100 | 1 | Mococo_Nsgaii |
| GaussianNB | 99.3 | 0.99 | Mococo_Nsgaii |
| XGBClassifier | 100 | 1 | Mococo_Nsgaii |
| DecisionTreeClassifier | 100 | 1 | Mococo_Nsgaii |
| LGBMClassifier | 100 | 1 | Mococo_Nsgaii |
| PassiveAggressiveClassifier | 100 | 1 | Mococo_Nsgaii |
| LogisticRegression | 100 | 1 | Mococo_Nsgaii |
| KNeighborsClassifier | 99.88 | 1 | Mococo_Nsgaii |
| SVC | 100 | 1 | Mococo_Nsgaii |
| SGDClassifier | 100 | 1 | Mococo_Nsgaii |
| ExtraTreesClassifier | 100 | 1 | Mococo_Nsgaii |
| CatBoost | 100 | 1 | Mococo_Nsgaii |
| RandomForestClassifier | 100 | 1 | CFS |
| GaussianNB | 98.48 | 0.98 | CFS |
| XGBClassifier | 100 | 1 | CFS |
| DecisionTreeClassifier | 100 | 1 | CFS |
| LGBMClassifier | 100 | 1 | CFS |
| PassiveAggressiveClassifier | 100 | 1 | CFS |
| LogisticRegression | 100 | 1 | CFS |
| KNeighborsClassifier | 100 | 1 | CFS |
| SVC | 100 | 1 | CFS |
| SGDClassifier | 100 | 1 | CFS |

| | | | |
|---|---|---|---|
| ExtraTreesClassifier | 100 | 1 | CFS |
| CatBoost | 98.48 | 0.98 | CFS |
| RandomForestClassifier | 100 | 1 | Consistency |
| GaussianNB | 77.96 | 0.77 | Consistency |
| XGBClassifier | 100 | 1 | Consistency |
| DecisionTreeClassifier | 100 | 1 | Consistency |
| LGBMClassifier | 100 | 1 | Consistency |
| PassiveAggressiveClassifier | 100 | 1 | Consistency |
| LogisticRegression | 99.84 | 1 | Consistency |
| KNeighborsClassifier | 100 | 1 | Consistency |
| SVC | 100 | 1 | Consistency |
| SGDClassifier | 100 | 1 | Consistency |
| ExtraTreesClassifier | 100 | 1 | Consistency |
| CatBoost | 100 | 1 | Consistency |

Mushroom Veri Kümesi En Yüksek Sınıflandırma Doğruluğu Sağlayan Algoritmalar

| | Selector | Accuracy |
|---|---|---|
| **RandomForestClassifier** | Mococo_Pareto | 100.00 |
| **RandomForestClassifier** | Mococo_Nsgaii | 100.00 |
| **RandomForestClassifier** | CFS | 100.00 |
| **RandomForestClassifier** | Consistency | 100.00 |

Primary Tumor Veri Kümesi Öznitelik Seçimi Sınıflandırma Sonuçları

| Model | Accuracy | F1 Score | Selector |
|---|---|---|---|
| RandomForestClassifier | 31.37 | 0.29 | Mococo_Pareto |
| GaussianNB | 22.55 | 0.19 | Mococo_Pareto |
| XGBClassifier | 38.24 | 0.36 | Mococo_Pareto |
| DecisionTreeClassifier | 32.35 | 0.31 | Mococo_Pareto |
| LGBMClassifier | 33.33 | 0.33 | Mococo_Pareto |
| PassiveAggressiveClassifier | 28.43 | 0.26 | Mococo_Pareto |

| | | | |
|---|---|---|---|
| LogisticRegression | 38.24 | 0.38 | Mococo_Pareto |
| KNeighborsClassifier | 27.45 | 0.24 | Mococo_Pareto |
| SVC | 38.24 | 0.33 | Mococo_Pareto |
| SGDClassifier | 36.27 | 0.33 | Mococo_Pareto |
| ExtraTreesClassifier | 31.37 | 0.3 | Mococo_Pareto |
| CatBoost | 35.29 | 0.34 | Mococo_Pareto |
| RandomForestClassifier | 34.31 | 0.32 | Mococo_Nsgaii |
| GaussianNB | 22.55 | 0.19 | Mococo_Nsgaii |
| XGBClassifier | 38.24 | 0.36 | Mococo_Nsgaii |
| DecisionTreeClassifier | 35.29 | 0.33 | Mococo_Nsgaii |
| LGBMClassifier | 33.33 | 0.33 | Mococo_Nsgaii |
| PassiveAggressiveClassifier | 27.45 | 0.26 | Mococo_Nsgaii |
| LogisticRegression | 38.24 | 0.38 | Mococo_Nsgaii |
| KNeighborsClassifier | 27.45 | 0.23 | Mococo_Nsgaii |
| SVC | 37.25 | 0.32 | Mococo_Nsgaii |
| SGDClassifier | 29.41 | 0.3 | Mococo_Nsgaii |
| ExtraTreesClassifier | 35.29 | 0.34 | Mococo_Nsgaii |
| CatBoost | 41.18 | 0.39 | Mococo_Nsgaii |
| RandomForestClassifier | 30.39 | 0.29 | CFS |
| GaussianNB | 23.53 | 0.19 | CFS |
| XGBClassifier | 38.24 | 0.36 | CFS |
| DecisionTreeClassifier | 31.37 | 0.29 | CFS |
| LGBMClassifier | 33.33 | 0.32 | CFS |
| PassiveAggressiveClassifier | 34.31 | 0.33 | CFS |
| LogisticRegression | 36.27 | 0.35 | CFS |
| KNeighborsClassifier | 32.35 | 0.31 | CFS |
| SVC | 36.27 | 0.32 | CFS |
| SGDClassifier | 30.39 | 0.3 | CFS |
| ExtraTreesClassifier | 34.31 | 0.32 | CFS |

| CatBoost | 32.35 | 0.3 | CFS |
|---|---|---|---|
| RandomForestClassifier | 31.37 | 0.29 | Consistency |
| GaussianNB | 22.55 | 0.19 | Consistency |
| XGBClassifier | 38.24 | 0.36 | Consistency |
| DecisionTreeClassifier | 32.35 | 0.31 | Consistency |
| LGBMClassifier | 33.33 | 0.33 | Consistency |
| PassiveAggressiveClassifier | 28.43 | 0.26 | Consistency |
| LogisticRegression | 38.24 | 0.38 | Consistency |
| KNeighborsClassifier | 27.45 | 0.24 | Consistency |
| SVC | 38.24 | 0.33 | Consistency |
| SGDClassifier | 36.27 | 0.33 | Consistency |
| ExtraTreesClassifier | 31.37 | 0.3 | Consistency |
| CatBoost | 35.29 | 0.34 | Consistency |

Primary Tumor Veri Kümesi En Yüksek Sınıflandırma Doğruluğu Sağlayan Algoritmalar

| | Selector | Accuracy |
|---|---|---|
| **CatBoost** | Mococo_Nsgaii | 41.18 |
| **XGBClassifier** | Mococo_Pareto | 38.24 |
| **XGBClassifier** | CFS | 38.24 |
| **XGBClassifier** | Consistency | 38.24 |

Soybean Veri Kümesi Öznitelik Seçimi Sınıflandırma Sonuçları

| Model | Accuracy | F1 Score | Selector |
|---|---|---|---|
| RandomForestClassifier | 90.22 | 0.9 | Mococo_Pareto |
| GaussianNB | 89.13 | 0.88 | Mococo_Pareto |
| XGBClassifier | 89.13 | 0.89 | Mococo_Pareto |
| DecisionTreeClassifier | 82.61 | 0.81 | Mococo_Pareto |
| LGBMClassifier | 90.22 | 0.9 | Mococo_Pareto |

| | | | |
|---|---|---|---|
| PassiveAggressiveClassifier | 88.04 | 0.87 | Mococo_Pareto |
| LogisticRegression | 91.3 | 0.91 | Mococo_Pareto |
| KNeighborsClassifier | 81.52 | 0.8 | Mococo_Pareto |
| SVC | 86.96 | 0.85 | Mococo_Pareto |
| SGDClassifier | 90.22 | 0.89 | Mococo_Pareto |
| ExtraTreesClassifier | 90.22 | 0.9 | Mococo_Pareto |
| CatBoost | 85.87 | 0.86 | Mococo_Pareto |
| RandomForestClassifier | 90.22 | 0.9 | Mococo_Nsgaii |
| GaussianNB | 90.22 | 0.89 | Mococo_Nsgaii |
| XGBClassifier | 86.96 | 0.86 | Mococo_Nsgaii |
| DecisionTreeClassifier | 83.7 | 0.82 | Mococo_Nsgaii |
| LGBMClassifier | 90.22 | 0.9 | Mococo_Nsgaii |
| PassiveAggressiveClassifier | 88.04 | 0.87 | Mococo_Nsgaii |
| LogisticRegression | 90.22 | 0.89 | Mococo_Nsgaii |
| KNeighborsClassifier | 82.61 | 0.82 | Mococo_Nsgaii |
| SVC | 86.96 | 0.85 | Mococo_Nsgaii |
| SGDClassifier | 91.3 | 0.92 | Mococo_Nsgaii |
| ExtraTreesClassifier | 91.3 | 0.91 | Mococo_Nsgaii |
| CatBoost | 92.39 | 0.92 | Mococo_Nsgaii |
| RandomForestClassifier | 90.22 | 0.9 | CFS |
| GaussianNB | 89.13 | 0.88 | CFS |
| XGBClassifier | 89.13 | 0.89 | CFS |
| DecisionTreeClassifier | 82.61 | 0.81 | CFS |
| LGBMClassifier | 90.22 | 0.9 | CFS |
| PassiveAggressiveClassifier | 88.04 | 0.87 | CFS |
| LogisticRegression | 91.3 | 0.91 | CFS |
| KNeighborsClassifier | 81.52 | 0.8 | CFS |
| SVC | 86.96 | 0.85 | CFS |
| SGDClassifier | 90.22 | 0.89 | CFS |

| | | | |
|---|---|---|---|
| ExtraTreesClassifier | 90.22 | 0.9 | CFS |
| CatBoost | 85.87 | 0.86 | CFS |
| RandomForestClassifier | 92.39 | 0.92 | Consistency |
| GaussianNB | 90.22 | 0.9 | Consistency |
| XGBClassifier | 89.13 | 0.89 | Consistency |
| DecisionTreeClassifier | 81.52 | 0.81 | Consistency |
| LGBMClassifier | 91.3 | 0.91 | Consistency |
| PassiveAggressiveClassifier | 88.04 | 0.88 | Consistency |
| LogisticRegression | 92.39 | 0.92 | Consistency |
| KNeighborsClassifier | 84.78 | 0.84 | Consistency |
| SVC | 85.87 | 0.85 | Consistency |
| SGDClassifier | 88.04 | 0.88 | Consistency |
| ExtraTreesClassifier | 89.13 | 0.88 | Consistency |
| CatBoost | 82.61 | 0.81 | Consistency |

Soybean Veri Kümesi En Yüksek Sınıflandırma Doğruluğu Sağlayan Algoritmalar

| | Selector | Accuracy |
|---|---|---|
| **RandomForestClassifier** | Consistency | 92.39 |
| **CatBoost** | Mococo_Nsgaii | 92.39 |
| **LogisticRegression** | Mococo_Pareto | 91.30 |
| **LogisticRegression** | CFS | 91.30 |

Spect Heart Veri Kümesi Öznitelik Seçimi Sınıflandırma Sonuçları

| Model | Accuracy | F1 Score | Selector |
|---|---|---|---|
| RandomForestClassifier | 73.75 | 0.72 | Mococo_Pareto |
| GaussianNB | 60 | 0.63 | Mococo_Pareto |
| XGBClassifier | 73.75 | 0.72 | Mococo_Pareto |
| DecisionTreeClassifier | 67.5 | 0.68 | Mococo_Pareto |
| LGBMClassifier | 76.25 | 0.76 | Mococo_Pareto |

| | | | |
|---|---|---|---|
| PassiveAggressiveClassifier | 67.5 | 0.6 | Mococo_Pareto |
| LogisticRegression | 73.75 | 0.72 | Mococo_Pareto |
| KNeighborsClassifier | 72.5 | 0.72 | Mococo_Pareto |
| SVC | 80 | 0.79 | Mococo_Pareto |
| SGDClassifier | 68.75 | 0.61 | Mococo_Pareto |
| ExtraTreesClassifier | 72.5 | 0.71 | Mococo_Pareto |
| CatBoost | 81.25 | 0.8 | Mococo_Pareto |
| RandomForestClassifier | 72.5 | 0.71 | Mococo_Nsgaii |
| GaussianNB | 60 | 0.63 | Mococo_Nsgaii |
| XGBClassifier | 73.75 | 0.72 | Mococo_Nsgaii |
| DecisionTreeClassifier | 65 | 0.66 | Mococo_Nsgaii |
| LGBMClassifier | 76.25 | 0.76 | Mococo_Nsgaii |
| PassiveAggressiveClassifier | 72.5 | 0.73 | Mococo_Nsgaii |
| LogisticRegression | 73.75 | 0.72 | Mococo_Nsgaii |
| KNeighborsClassifier | 73.75 | 0.74 | Mococo_Nsgaii |
| SVC | 75 | 0.75 | Mococo_Nsgaii |
| SGDClassifier | 73.75 | 0.74 | Mococo_Nsgaii |
| ExtraTreesClassifier | 72.5 | 0.71 | Mococo_Nsgaii |
| CatBoost | 78.75 | 0.77 | Mococo_Nsgaii |
| RandomForestClassifier | 75 | 0.75 | CFS |
| GaussianNB | 60 | 0.63 | CFS |
| XGBClassifier | 73.75 | 0.72 | CFS |
| DecisionTreeClassifier | 65 | 0.66 | CFS |
| LGBMClassifier | 76.25 | 0.76 | CFS |
| PassiveAggressiveClassifier | 70 | 0.71 | CFS |
| LogisticRegression | 73.75 | 0.72 | CFS |
| KNeighborsClassifier | 68.75 | 0.69 | CFS |
| SVC | 76.25 | 0.76 | CFS |
| SGDClassifier | 75 | 0.75 | CFS |

| | | | |
|---|---|---|---|
| ExtraTreesClassifier | 72.5 | 0.72 | CFS |
| CatBoost | 77.5 | 0.78 | CFS |
| RandomForestClassifier | 75 | 0.74 | Consistency |
| GaussianNB | 60 | 0.63 | Consistency |
| XGBClassifier | 73.75 | 0.72 | Consistency |
| DecisionTreeClassifier | 70 | 0.7 | Consistency |
| LGBMClassifier | 76.25 | 0.76 | Consistency |
| PassiveAggressiveClassifier | 67.5 | 0.6 | Consistency |
| LogisticRegression | 73.75 | 0.72 | Consistency |
| KNeighborsClassifier | 73.75 | 0.74 | Consistency |
| SVC | 78.75 | 0.78 | Consistency |
| SGDClassifier | 68.75 | 0.61 | Consistency |
| ExtraTreesClassifier | 73.75 | 0.73 | Consistency |
| CatBoost | 78.75 | 0.77 | Consistency |

Spect Heart Veri Kümesi En Yüksek Sınıflandırma Doğruluğu Sağlayan Algoritmalar

| | Selector | Accuracy |
|---|---|---|
| **CatBoost** | Mococo_Pareto | 81.25 |
| **CatBoost** | Mococo_Nsgaii | 78.75 |
| **SVC** | Consistency | 78.75 |
| **CatBoost** | CFS | 77.50 |

Splice Veri Kümesi Öznitelik Seçimi Sınıflandırma Sonuçları

| Model | Accuracy | F1 Score | Selector |
|---|---|---|---|
| RandomForestClassifier | 91.54 | 0.91 | Mococo_Pareto |
| GaussianNB | 51.93 | 0.49 | Mococo_Pareto |
| XGBClassifier | 95.51 | 0.96 | Mococo_Pareto |
| DecisionTreeClassifier | 93.1 | 0.93 | Mococo_Pareto |

| | | | |
|---|---|---|---|
| LGBMClassifier | 96.03 | 0.96 | Mococo_Pareto |
| PassiveAggressiveClassifier | 94.15 | 0.94 | Mococo_Pareto |
| LogisticRegression | 95.19 | 0.95 | Mococo_Pareto |
| KNeighborsClassifier | 34.27 | 0.22 | Mococo_Pareto |
| SVC | 74.71 | 0.74 | Mococo_Pareto |
| SGDClassifier | 69.38 | 0.65 | Mococo_Pareto |
| ExtraTreesClassifier | 92.58 | 0.92 | Mococo_Pareto |
| CatBoost | 98.01 | 0.98 | Mococo_Pareto |
| RandomForestClassifier | 90.8 | 0.91 | Mococo_Nsgaii |
| GaussianNB | 45.98 | 0.44 | Mococo_Nsgaii |
| XGBClassifier | 95.09 | 0.95 | Mococo_Nsgaii |
| DecisionTreeClassifier | 92.16 | 0.92 | Mococo_Nsgaii |
| LGBMClassifier | 94.98 | 0.95 | Mococo_Nsgaii |
| PassiveAggressiveClassifier | 93.31 | 0.93 | Mococo_Nsgaii |
| LogisticRegression | 94.78 | 0.95 | Mococo_Nsgaii |
| KNeighborsClassifier | 32.92 | 0.21 | Mococo_Nsgaii |
| SVC | 75.55 | 0.75 | Mococo_Nsgaii |
| SGDClassifier | 67.82 | 0.63 | Mococo_Nsgaii |
| ExtraTreesClassifier | 88.92 | 0.89 | Mococo_Nsgaii |
| CatBoost | 98.75 | 0.99 | Mococo_Nsgaii |
| RandomForestClassifier | 97.6 | 0.98 | CFS |
| GaussianNB | 73.77 | 0.64 | CFS |
| XGBClassifier | 98.43 | 0.98 | CFS |
| DecisionTreeClassifier | 96.34 | 0.96 | CFS |
| LGBMClassifier | 99.06 | 0.99 | CFS |
| PassiveAggressiveClassifier | 97.91 | 0.98 | CFS |
| LogisticRegression | 96.45 | 0.96 | CFS |
| KNeighborsClassifier | 74.29 | 0.75 | CFS |
| SVC | 96.55 | 0.97 | CFS |

| | | | |
|---|---|---|---|
| SGDClassifier | 96.76 | 0.97 | CFS |
| ExtraTreesClassifier | 96.34 | 0.96 | CFS |
| CatBoost | 93.21 | 0.93 | CFS |
| RandomForestClassifier | 86.42 | 0.86 | Consistency |
| GaussianNB | 32.18 | 0.2 | Consistency |
| XGBClassifier | 95.4 | 0.95 | Consistency |
| DecisionTreeClassifier | 91.12 | 0.91 | Consistency |
| LGBMClassifier | 95.4 | 0.95 | Consistency |
| PassiveAggressiveClassifier | 78.79 | 0.79 | Consistency |
| LogisticRegression | 77.12 | 0.76 | Consistency |
| KNeighborsClassifier | 31.03 | 0.2 | Consistency |
| SVC | 47.96 | 0.38 | Consistency |
| SGDClassifier | 63.01 | 0.57 | Consistency |
| ExtraTreesClassifier | 80.88 | 0.8 | Consistency |
| CatBoost | 86.62 | 0.87 | Consistency |

Splice Veri Kümesi En Yüksek Sınıflandırma Doğruluğu Sağlayan Algoritmalar

| | Selector | Accuracy |
|---|---|---|
| **LGBMClassifier** | CFS | 99.06 |
| **CatBoost** | Mococo_Nsgaii | 98.75 |
| **CatBoost** | Mococo_Pareto | 98.01 |
| **XGBClassifier** | Consistency | 95.40 |

Voting Records Veri Kümesi Öznitelik Seçimi Sınıflandırma Sonuçları

| Model | Accuracy | F1 Score | Selector |
|---|---|---|---|
| RandomForestClassifier | 99.23 | 0.99 | Mococo_Pareto |
| GaussianNB | 93.85 | 0.94 | Mococo_Pareto |
| XGBClassifier | 97.69 | 0.98 | Mococo_Pareto |
| DecisionTreeClassifier | 95.38 | 0.95 | Mococo_Pareto |
| LGBMClassifier | 97.69 | 0.98 | Mococo_Pareto |

| | | | |
|---|---|---|---|
| PassiveAggressiveClassifier | 94.62 | 0.95 | Mococo_Pareto |
| LogisticRegression | 96.92 | 0.97 | Mococo_Pareto |
| KNeighborsClassifier | 94.62 | 0.95 | Mococo_Pareto |
| SVC | 98.46 | 0.98 | Mococo_Pareto |
| SGDClassifier | 97.69 | 0.98 | Mococo_Pareto |
| ExtraTreesClassifier | 99.23 | 0.99 | Mococo_Pareto |
| CatBoost | 99.23 | 0.99 | Mococo_Pareto |
| RandomForestClassifier | 99.23 | 0.99 | Mococo_Nsgaii |
| GaussianNB | 95.38 | 0.95 | Mococo_Nsgaii |
| XGBClassifier | 97.69 | 0.98 | Mococo_Nsgaii |
| DecisionTreeClassifier | 95.38 | 0.95 | Mococo_Nsgaii |
| LGBMClassifier | 97.69 | 0.98 | Mococo_Nsgaii |
| PassiveAggressiveClassifier | 93.85 | 0.94 | Mococo_Nsgaii |
| LogisticRegression | 96.92 | 0.97 | Mococo_Nsgaii |
| KNeighborsClassifier | 96.15 | 0.96 | Mococo_Nsgaii |
| SVC | 99.23 | 0.99 | Mococo_Nsgaii |
| SGDClassifier | 98.46 | 0.98 | Mococo_Nsgaii |
| ExtraTreesClassifier | 99.23 | 0.99 | Mococo_Nsgaii |
| CatBoost | 98.46 | 0.98 | Mococo_Nsgaii |
| RandomForestClassifier | 99.23 | 0.99 | CFS |
| GaussianNB | 97.69 | 0.98 | CFS |
| XGBClassifier | 97.69 | 0.98 | CFS |
| DecisionTreeClassifier | 95.38 | 0.95 | CFS |
| LGBMClassifier | 97.69 | 0.98 | CFS |
| PassiveAggressiveClassifier | 95.38 | 0.95 | CFS |
| LogisticRegression | 96.92 | 0.97 | CFS |
| KNeighborsClassifier | 97.69 | 0.98 | CFS |
| SVC | 98.46 | 0.98 | CFS |
| SGDClassifier | 97.69 | 0.98 | CFS |

| | | | |
|---|---|---|---|
| ExtraTreesClassifier | 99.23 | 0.99 | CFS |
| CatBoost | 97.69 | 0.98 | CFS |
| RandomForestClassifier | 98.46 | 0.98 | Consistency |
| GaussianNB | 95.38 | 0.95 | Consistency |
| XGBClassifier | 97.69 | 0.98 | Consistency |
| DecisionTreeClassifier | 95.38 | 0.95 | Consistency |
| LGBMClassifier | 97.69 | 0.98 | Consistency |
| PassiveAggressiveClassifier | 93.85 | 0.94 | Consistency |
| LogisticRegression | 96.92 | 0.97 | Consistency |
| KNeighborsClassifier | 97.69 | 0.98 | Consistency |
| SVC | 99.23 | 0.99 | Consistency |
| SGDClassifier | 97.69 | 0.98 | Consistency |
| ExtraTreesClassifier | 99.23 | 0.99 | Consistency |
| CatBoost | 98.46 | 0.98 | Consistency |

Voting Records Veri Kümesi En Yüksek Sınıflandırma Doğruluğu Sağlayan Algoritmalar

| | Selector | Accuracy |
|---|---|---|
| **CatBoost** | Mococo_Pareto | 99.23 |
| **RandomForestClassifier** | Mococo_Nsgaii | 99.23 |
| **RandomForestClassifier** | CFS | 99.23 |
| **SVC** | Consistency | 99.23 |

# ÖZGEÇMİŞ

| Kişisel Bilgiler | |
|---|---|
| Adı Soyadı | Emrah SEZER |
| Doğum Yeri | |
| Doğum Tarihi | Tarih girmek için tıklayın veya dokunun. |
| Uyruğu | ☑ T.C. ❒ Diğer: |
| E-Posta Adresi | |
| Web Adresi | |

| Eğitim Bilgileri<br>Lisans | |
|---|---|
| Üniversite | Kocaeli Üniversitesi |
| Fakülte | Fen-Edebiyat Fakültesi |
| Bölümü | Matematik Bölümü |
| Mezuniyet Yılı | |

| Yüksek Lisans | |
|---|---|
| Üniversite | Marmara Üniversitesi |
| Enstitü Adı | Sosyal Bilimler Enstitüsü |
| Anabilim Dalı | İşletme Anabilim Dalı |
| Programı | Sayısal Yöntemler Programı |

| Doktora | |
|---|---|
| Üniversite | İstanbul Üniversitesi |
| Enstitü Adı | Fen Bilimleri Enstitüsü |
| Anabilim Dalı | Enformatik Anabilim Dalı |
| Programı | Enformatik Programı |

**Makale ve Bildiriler**

Ayaz, T. B., Sezer, E., Çelik, A. E. (2023) Potansiyel müşteri segmentasyonu uygulaması, *2. Uluslararası İktisadi ve İdari Bilimler Kongresi,* 19-20 Ocak 2023, Bingöl Üniversitesi, 187-188.

Başeğmez, H., Sezer, E. ve Erol, Ç. S. (2021). Optimization for gene selection and cancer classification. *MDPI Proceedings,* 74(21), 1-10. https://doi.org/10.3390/proceedings2021074021

Sezer, E. ve Başeğmez, H. (2022). An Approach Based on Feature Selection for Missing Value Imputation. In *Intelligent and Fuzzy Techniques for Emerging*

*Conditions and Digital Transformation: Proceedings of the INFUS 2021 Conference, August 24-26, 2021. Volume 1* (945-950). Springer, Cham.

Sezer, E. ve Çakır, Ö. (2022). Sınıflandırma Amaçlı Değişken Alt Kümesi Seçimi: Bir Bankacılık Uygulaması. *Dicle Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 12 (24), 480-498. Doi: 10.53092/duiibfd.1137444