




**T.C.**
**SELÇUK ÜNİVERSİTESİ**
**FEN BİLİMLERİ ENSTİTÜSÜ**

# FARKLI VERİ SETLERİ ÜZERİNDE ATEŞBÖCEĞİ OPTİMİZASYON ALGORİTMASI İLE KÜMELEME

**Mina Muayad Abdulazeez ALABD ALRAHMAN**

**YÜKSEK LİSANS TEZİ**

**Bilgisayar Mühendisliği Anabilim Dalı**

**Şubat 2019**
**KONYA**
**Her Hakkı Saklıdır**

# ÖZET

## YÜKSEK LİSANS TEZİ

## FARKLI VERİ SETLERİ ÜZERİNDE ATEŞBÖCEĞİ OPTİMİZASYON ALGORİTMASI İLE KÜMELEME

**Mina Muayad Abdulazeez ALABD ALRAHMAN**

**Selçuk Üniversitesi Fen Bilimleri Enstitüsü**
**Bilgisayar Mühendisliği Anabilim Dalı**

**Danışman: Doç. Dr. Hasan Erdinç KOÇER**

**2019, 37 Sayfa**

**Jüri**
Doç. Dr. Hasan Erdinç KOÇER
Prof. Dr. Harun UĞUZ
Doç. Dr. Gülay TEZEL

Kümeleme denetimsiz bir sınıflandırma algoritmasıdır ve nesneler arasında bulunan benzerliğe göre işlem yapar. Aynı kümede bulunan nesneler çok benzerlik gösterirken diğer kümeler arasında farklılık göstermektedir. Kümeleme işlemi birçok kümeleme algoritmaları ile yapılabilir, önemli olan veriler arasında en iyi küme merkezlerinin bulunmasıdır. Bu çalışmada global arama yeteneğine sahip olan ve birçok zor problemlerin çözümünde kullanılan ateşböceği kümeleme algoritması kullanılarak UCI veri ambarından alınan 12 adet veri seti (Balance, Breast Cancer Wisconsin Diagnostic, Breast Cancer Wisconsin Original, Credit, Dermatology, Pima Diabetes, E. Coli, Glass, Heart Disease, İris, Newthyroid ve Wine) üzerinde kümeleme işlemi yapılmıştır. Tezde önerilen yaklaşımda Rand index fonksiyonu kümeleme aşamasında uygunluk fonksiyonu olarak kullanılmıştır. Önerilen kümeleme algoritmasının test sonuçları, aynı veri setlerine uygulanan Karışık Kurbağa Sıçrama Algoritması (KKSA) ve Parçacık Sürü Optimizasyonu (PSO) sonuçları (Karakoyun, 2015) (Karakoyun ve Babalik, 2015) ile karşılaştırılmıştır. Bu karşılaştırmaya göre önerilen kümeleme algoritması 7 veri setinde (Balance, Breast Cancer Wisconsin Original, E.Coli, Glass, Heart Disease, İris ve Wine) daha iyi bir performans göstermiştir. Bu karşılaştırmalarda ölçüt olarak sınıflandırma hata yüzdesi (SHY) hesaplanmış ve önerilen algoritmadan elde edilen performanslar tezin deneysel sonuçlar kısmında karşılaştırmalı olarak sunulmuştur.

**Anahtar Kelimeler:** Ateşböceği algoritması, kümeleme, optimizasyon algoritması.

# ABSTRACT

## MS THESIS

## CLUSTERING OF FIREFLY OPTIMIZATION ALGORITHM ON DIFFERENT DATA SETS

**Mina Muayad Abdulazeez ALABD ALRAHMAN**

**THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCE OF SELÇUK UNIVERSITY**
**THE DEGREE OF MASTER OF SCIENCE OF PHILOSOPHY**
**IN COMPUTER ENGINEERING**

**Advisor: Assoc. Prof. Dr. Hasan Erdinç Koçer**

**2019, 37 Pages**

**Jury**
Advisor: Assoc. Prof. Dr. Hasan Erdinç KOÇER
Prof. Dr. Harun UĞUZ
Advisor: Assoc. Prof. Dr. Gülay TEZEL

Clustering is an unsupervised classification algorithm, is a group of clustering method, the clusters in the same group are very similar and the clusters in the other group are different. This clustering can be done with many clustering algorithms, it is important to find the best cluster centers among the data. In this thesis, the success of the firefly clustering algorithm, which has global search ability and is used to solve many difficult problems, has been tested on 12 datasets (Balance, Breast Cancer Wisconsin Diagnostic, Breast Cancer Wisconsin Original, Credit, Dermatology, Pima Diabetes, E. Coli, Glass, Heart Disease, Iris, Newthyroid and Wine) from the frequently used UCI data warehouse. In the proposed approach, the Rand index function was used as a fitness function. The test results of the proposed clustering algorithm were compared with the results of Suffled Frog Leaping Algorithm (SFLA) application and Particle Swarm Optimization (PSO) results (Karakoyun, 2015 ) (Karakoyun and Babalik, 2015). According to this comparison, the proposed clustering algorithm showed better performance in 7 data sets (Balance, Breast Cancer, Wisconsin Original, E.Coli, Glass, Heart Disease, Iris and Wine). In this comparison, Classification Error Percentage (CEP) and the performances obtained from the proposed algorithm were presented comparatively in the experimental results section of the thesis.

**Keywords:** Firefly algorithm, clustering, optimization algorithm.

# ÖNSÖZ

Yüksek lisans öğrenim sürecimin tüm aşamalarında engin bilgi birikimini, deneyimini ve ilgisini hiçbir zaman esirgemeyen değerli danışmanım ve hocam Sayın Doç. Dr. Hasan Erdinç KOÇER'e ve sayın Mohammed H. IBRAHIM'e hocalarıma sonsuz teşekkürlerimi sunarım.

Eğitim ve öğrenim hayatım boyunca maddi ve manevi yönden beni her zaman destekleyen, üzerimde büyük hakları olan aileme teşekkürü bir borç bilirim

Mina Muayad Abdulazeez ALABD ALRAHMAN
KONYA-2019

# İÇİNDEKİLER

**ÖZET** ..... **iv**

**ABSTRACT** ..... **v**

**ÖNSÖZ** ..... **vi**

**İÇİNDEKİLER** ..... **vii**

**KISALTMALAR** ..... **viii**

**1. GİRİŞ** ..... **1**

**2. KAYNAK ARAŞTIRMASI** ..... **3**

**3. MATERYAL VE YÖNTEM** ..... **11**

3.1. Test (Benchmark) problemleri ..... 11
3.2. Kümeleme ve Kümeleme Yöntemleri ..... 13
3.2.1. Bölümleme tabanlı kümeleme algoritması ..... 13
3.2.2. Hiyerarşik tabanlı kümeleme algoritması ..... 14
3.2.3. Yoğunluk tabanlı kümeleme algoritması ..... 15
3.2.4. Bulanık mantık kümeleme algoritması ..... 16

**4. OPTİMİZASYON** ..... **18**

4.1. Ateşböceği Optimizasyon Algoritması ..... 18

**5. ATEŞBÖCEĞİ ALGORİTMASI İLE KÜMELEME** ..... **22**

5.1. Kullanılan Yöntem ..... 22
5.2. Uygulama Sonuçları ..... 24

**6. SONUÇ** ..... **32**

**KAYNAKLAR** ..... **34**

**ÖZGEÇMİŞ** ..... **37**

## KISALTMALAR

| | |
|---|---|
| AA: | Ateşböceği Algoritması |
| FA: | Firefly Algorithm |
| KKSA: | Karışık Kurbağa Sıçrama Algoritması |
| SFLA: | Shuffled Frog Leaping Algorithm |
| SHY: | Sınıflandırma Hata Yüzdesi (Classification Error Percentage CEP) |

# 1. GİRİŞ

Kümeleme, veri madenciliğinde yaygın şekilde kullanılan verileri gruplandırmak için yapılan bir işlemdir (Kayım, 2015 ). Kümeleme, veri analizinin önemli adımlarından birisi olarak yaygın şekilde araştırmacılar tarafından kullanılmaktadır. Bu algoritmalar, çoğunlukla tanımlanmamış sınıf gruplarını bulmak için kullanılır (Gulati ve Singh, 2015). Burada her kümenin diğer küme verileri ile büyük benzerlik ve yüksek benzerlik olmayan verilere sahip olması gerekir (Rana ve ark., 2013). Seçilen metriği bağlı olarak bir veri nesnesi tek bir kümeye ait olabilir veya birden fazla kümeye ait olabilir (Gulati ve Singh, 2015). Böylece aynı gruptaki noktalar birbirine benzerse de farklı gruplardaki noktalar birbirine benzemez. Bu temel kural, kümeleme algoritmalarının tasarımını ve kümelenmelerin değerlendirilmesini yönlendirir. Mevcut parametrik kümeleme yöntemlerinin çoğu, tanımlanmış bir maliyet fonksiyonunun en aza indirgenmesi yaklaşımı ile veriyi ve temsilcilerini önceden tanımlanmış sayıda küme haline getirmeyi amaçlar. Bu nedenle, kümeleme ile ilgili üç husus vardır. Bunlar: veriler, maliyet fonksiyonu ve değerlendirme fonksiyonu (Zhao, 2012).

Bu çalışmada, global arama yeteneği olan ve birçok zor problemlerin çözümünde tercih edilen ateşböceği optimizasyon algoritması (Firefly Algorithm (FA)), optimum küme merkezlerini bulmak için kullanılmıştır. Ateşböceği algoritması (AA), tropikal iklim bölgelerinde yaşayan ateşböceklerinin sosyal davranışlarını baz alan bir metasezgisel optimizasyon algoritmasıdır. Bu algoritmada ateşböceklerinin diğer ateşböcekleri ile haberleşmesini sağlayan ışık yayma davranışı modellenerek optimizasyon işlemine uyarlanması gerçekleştirilmektedir (Yang, 2009). AA içerisinde uygunluk fonksiyonu olarak çalışan Rand Index ile yapılan kümeleme işleminin sonuçları irdelenmiştir. Önerilen kümeleme algoritması literatürde sıkça kullanılan test (benchmark) problemleri üzerinde test edilmiştir. Bu veri setleri UCI Makine Öğrenim ambarından alınan 12 adet veri kümesini içermektedir.

Çalışmanın birinci bölümünde kümeleme problemi üzerine bir giriş yapılmış ve kümelemenin tanımından yola çıkılarak AA optimizasyon algoritması hakkında bilgi verilmiştir.

Çalışmanın ikinci bölümünde tez konusu ile ilgili yapılmış önceki çalışmalara değinilmiş ve çalışmaların sonuçları hakkında kısa bilgi verilmiştir.

Çalışmanın üçüncü bölümünde uygulamada kullanılan materyal ve yöntem anlatılmıştır. Materyal olarak kullanılan test veri kümeleri verilmiş ve ardından kümeleme, sınıflandırma ve optimizasyon işlemlerine değinilmiştir.

Çalışmanın dördüncü bölümünde önerilen yaklaşım çerçevesinde Rand Index uygunluk fonksiyonuna sahip ateşböceği kümeleme algoritması anlatılmıştır. Bölüm sonunda test veri setlerinden alınan uygulama sonuçları verilmiş ve Karışık Kurbağa Sıçrama Algoritması (Shuffled Frog Leaping Algorithm) ve Parçacık Sürü Optimizasyonu (Particle Swarm Optimization) ile elde edilen test sonuçları (Karakoyun, 2015 ) (Karakoyun ve Babalik, 2015) ile karşılaştırılmıştır.

Çalışmanın beşinci ve son bölümünde ise tez kapsamında elde edilen sonuçlara değinilerek önerilerde bulunulmuştur.

## 2. KAYNAK ARAŞTIRMASI

Kaynak araştırmasında verilen çalışmalar tarih sırasına göre kümeleme çalışmaları, optimizasyon çalışmaları ve optimizasyon algoritması kullanılarak kümeleme şeklinde gruplandırılmıştır.

(Keleş, 2017), tezinde K-ortalama kümeleme algoritması kullanılarak Türkiye Kamu Hastaneleri Kurumu'na bağlı verimlilik karnesi hesaplanmıştır. Hastanelerin kümelenmesi çalışmasına odaklanılmıştır. Hastaneler; mali durum, cihaz kapasitesi, personel kapasitesi ve ürettiği tıbbi hizmetlerin hacmi ve çeşitliliği dikkate alınarak kümelenmiştir. Yapılan kümeleme çalışması sonucunda; verimlilik karne uygulaması tarafından değerlendirilecek 597 hastane 16 kümeye ayrılmıştır. Kümeleme algoritmasında girdi olarak kullanılan hastane veri seti için 149 nitelik belirlenmiştir. Kümeleme çalışması verilerin uygulama kaynaklarından sistematik olarak çekilmesi ve doğru sıra dışı hastanelerin tespiti de dikkate alınarak büyük ölçüde başarılı bulunmuştur.

(Aderinwale, 2017), tezinde, K-ortalama algoritmasıyla üç farklı veri tipini (gen ifadesi, DNA metilasyon, miRNA) entegre etmiştir ve bunu iki farklı kanser verisinde uygulamıştır. Farklı veri tiplerinden elde edilen çekirdek matrislerini öğrenilen ağırlıklarla birleştirmenin, bu çekirdekleri eş oranlarla birleştirmekten ya da veri tiplerini bağımsız olarak kullanmaktan daha iyi sonuç verdiği ifade edilmiştir. Elde edilen kümelerin biyolojik anlamlılığını göstermek için Kaplan-Meier analizini uyguladığında kümelerin istatistiksel olarak farklı sağlam profilleri göstermiştir. Daha sonra genomik verilerin çok-boyutluluk problemini hedefleyerek k-ortalama algoritmasını çok-tipli verilere uygulanacak şekilde geliştirmiştir. Geliştirilen yöntemde hem öznitelikler için hem de veri tipleri için ağırlıklar öğrenilmekte, böylece tamamlayıcı kümeleme ve öznitelik seçimi aynı anda gerçekleştirilebilmektedir. Yaptığı deneylerin büyük çoğunluğunda öznitelik seçiminin diğer yöntemlerden daha iyi sonuç verdiği belirtilmiştir. Sonuçlar, her iki bölümden elde edilen sonuçlar farklı genomik veri tiplerinin birleştirilmesinin ve öznitelik seçiminin uygulanması ile elde edilen alt grupların var olan alt gruplara göre daha iyi olmuştur.

(Amin, 2016), tezinde düşük enerji uyumlu hiyerarşik kümeleme protokolü ve K-ortalama mimarisini birleştiren küme yapısı gerçekleştirmiştir. Ayrıca, eşik değere bağlı ve birbirine yakın mesafedeki düğümleri hedef alarak ikinci düzey bir kümeleme gerçekleştirilmiş ve enerji verimli yakın düğümleri kümeleme yöntemi geliştirilmiştir. Hedef rastgele dağılımda gerçekleşen ve aynı algı mesafesi kapsamındaki alanda birden

fazla algılayıcıdan ziyade tek bir algılayıcının çalışması ve algılama şeklindedir. Böylece algılama esnasında, aynı verilerin tekrarlanarak toplanmasının engellemesinden dolayı, paket hacminin azalması ve algılama için harcanan enerjinin azaltılması yoluyla enerji tasarrufu sağlanmıştır.

(Yıldız, 2015), çalışmasında K-ortalama ve SVM algoritmaları kullanarak Türkiye genelinde il ve yıl bazında vatandaşların evlenme, boşanma, sinema izleme ve tiyatro izleme yoğunluklarının birbirine etkisini sosyal durum analizi olarak gözlemlemiştir. Buna göre bir bölgedeki istatistikler birbirlerini olumlu ya da olumsuz etkilemektedir.

(Demirkan, 2014), tezinde k-ortalama algoritması ve Jaccard benzerlik ölçütünü kullanarak kategorik ve kategorik olmayan veriler üzerinde çalışma yapmıştır. Çalışmada kategorik ve kategorik olmayan karışık veri kümeleri alınarak, ne oranda anlamlı, mantıklı ve doğru bir sonuç kümesi oluşturulabileceğine yönelik analiz gerçekleştirilmiştir. Kategorik olmayan veri kümeleri için K-Means algoritması, kategorik veri kümeleri içinse Jaccard benzerlik ölçütünden yararlanılmış ve iki kümeleme yöntemi birleştirilip örnek veri setleri kullanılarak yepyeni anlamlı bir sonuç kümesi, kümeleme yöntemi oluşturulmaya çalışılmıştır.

(Khalaf, 2014), çalışmasında kablosuz algılayıcı ağların enerji verimli ve bağlantı sorununu incelemiştir. Küme başlarının yakın düğümleri ve uzak düğümleri arasında enerji düzeyleri açısından büyük farklar bulunmuştur. Tüm ağ alanını eşit alana bölerek bu sorun telafi edilecek ve her bölüm için farklı kümelenme politikaları uygulanmıştır. Sonra düşük enerji uyarlamalı kümeleme hiyerarşi ile protokol performansını karşılaştırmıştır. Önerilen sistemin performansının önceki çalışmalardan daha iyi olduğu gösterilmiştir. Gerçekleştirilen protokolde yoğun trafik olan ağlarda veri iletiminde enerji tüketim miktarının ve paket kaybının indirgenmesi garantilenmiştir.

(Şişeci, 2012), tezinde öğreticisiz öğrenme modelleri üzerinde durulmuş, traverten plakalar için yeni bir sınıf tanımlaması öngörülmüştür. Bu kapsamda birçok öznitelik analizi yapılarak traverten plaka taşlar için en uygun değeri veren öznitelikler belirlenmiştir. Öğreticisiz öğrenme tekniklerinden K-ortalamalar kümeleme yöntemi bölümleme yöntemiyle hızlandırılmaya çalışmıştır. Yapılan deneysel sonuçlar, önerilen yöntem ile traverten sınıflarının ayrımında, hızlandırılmış K-ortalamalar kümeleme yöntemi ile daha hızlı ve belirginmiş ayrıca sınıf içi dağılımlarının göze daha hoş geldiğini göstermiştir.

(Nasibov, 2012), tezinde hiyerarşik olmayan kümeleme yaklaşımına dayanan, veri setindeki veri grupları arasında kesin ayrımın söz konusu olduğu, kesin kümeleme

algoritmalarından olan artımlı algoritmalar incelemiştir. Kesin kümeleme içerisinde yer alan artımlı algoritmalar, veri setindeki veri grupları arasında kesin ayrımın söz konusu olmadığı, bir verinin belirli bir üyelik derecesiyle birden fazla kümeye ait olabildiği bulanık kümeleme yöntemleri ile beraber ele alınıp, bulanık kümeleme problemleri için yeni bir algoritma önermiştir. Önerilen artımlı yöntem 12 gerçek veri seti üzerinde uygulanmış ve Bulanık C-ortalamalar algoritması ile kıyaslandığında yöntemin yararlılığı gösterilmiştir.

(Turel ve Can, 2011), çalışmalarında arama motorları tarafından oluşturulan uzun doküman listesini, gruplanmış ve etiketlenmiş kümelere ayıran yeni bir arama sonucu bulma yaklaşımı sunmuşlardır. Bu yaklaşımda kapsama katsayısına dayalı kümeleme ve sıralı k-ortalamalar algoritmaları kullanılarak kümeleme kalitesine önem verilmiştir. Kümelerin etiketlemesi, anlamsız ya da kafa karıştıran etiketlerin kullanıcıları yanlış kümelere yönlendirerek zaman kaybettirmesi nedeniyle önemlidir. Bunlara ek olarak, bir kümenin etiketi, kümede bulunan dokümanların içeriklerini doğru bir biçimde yansıtmalıdır. Kümeleri etiketleme görevini etkin bir şekilde yerine getirebilmek için, terim ağırlıklarına dayalı yeni bir küme etiketleme yöntemi sunmuştur. Ayrıca küme etiketlemenin başarısını değerlendirmek için hassasiyet ve kesinlik ölçütlerini kullanan yeni bir etiketleme metriği sunmuştur. Metodu Sonek Ağacıyla Kümeleme ve Lingo gibi önde gelen arama sonucu kümeleme algoritmalarına göreceli performansını saptayabilmek için karşılaştırmalı bir değerlendirme yöntemi uygulanmıştır. Diğer taraftan, herkesin kullanımına açık olan Ambient ve ODP-239 veri setlerinde testler gerçekleştirilmiştir. Test sonuçları önerilen metodun hem kümeleme hem de etiketleme görevini başarıyla yerine getirmiştir.

(Erdil, 2011), tezinde çoklu kümelemelerin birleştirilmesi için çizge tabanlı, ölçeklenebilir, güçlü ve sezgisel bir algoritma önermiştir. Çoklu kümelemelerin birleştirilmesi, önceden var olan bilgiyi yeniden kullanmayı ve daha iyi bir kaliteye sahip olan yeni bir sonuç kümelemeyi üretmeyi gerektirir. Önerilen algoritma, nesnelerden oluşan, ağırlıklı ve girdi kümelenmelerindeki kanıt biriktirilerek oluşturulmuş bir benzerlik çizgesi üzerinde çalışır. Çalışmada gerçek ve sanal olarak üretilmiş ve gen ifade eden zorlayıcı veri kümeleri üzerindeki deneysel sonuçlar elde edilmiş ve yöntemin hem kalite hem de çalışma zamanı açısından kullanışlı olduğu ifade edilmiştir.

(Wang ve ark., 2017), komşuluk cazibeli ateşböceği algoritması olarak adlandırılan yeni bir algoritma önermişlerdir. Sıkça tercih edilen bazı karşılaştırma fonksiyonları kullanılarak deneysel çalışma gerçekleştirilmiştir. Sonuçlar, önerilen

yöntemin, çözümlerin doğruluğunu etkin bir şekilde artırabildiğini ve hesaplama zaman karmaşıklığını azaltabildiğini göstermiştir.

(A. J. Umbarkar, 2017), bu makalede önerilen hızlı sıralama, ateşböceğinin zaman karmaşıklığını azaltmak için kabarcık sıralamasını değiştirmektedir. Kullanılan veri kümesi, CEC 2005'in kısıtlanmamış kıyaslama fonksiyonudur. Hızlı sıralama ile ateşböceğinin türünü ve ateşböceği kullanarak ateşböceği karşılaştırması, en iyi, en kötü, ortalama, standart sapma, karşılaştırma sayısı ve yürütme süresine göre gerçekleştirilmiştir. Deneme sonucu, hızlı sıralama kullanan ateşböceğinin daha az sayıda karşılaştırma gerektirdiğini, ancak daha fazla yürütme süresi gerektirdiğini göstermiştir.

(Pamuk, 2016), bu makalede termik güç santralleri etkisindeki ekonomik yük dağıtım problemleri kaos teorisine dayalı Kaotik Ateşböceği Optimizasyon Algoritması (KAOA) ile çözülmüştür. Ağırlıklı toplam metodu kullanılarak çok amaçlı optimizasyon problemi olarak tanımlanan termik güç santralleri etkisindeki ekonomik yük dağıtım problemleri, tek amaçlı optimizasyon problemlerine çevrilmiştir. Kayıp matrisi, problemin çözümü için gerekli olan enerji iletim hatlarındaki hat kayıplarını hesaplamak için kullanılmıştır. Elektriksel sınırlamalar dikkate alınarak, toplam yakıt maliyeti minimize edilmiştir. Ekonomik yük dağıtım problemleri IEEE-57 baralı 7 jeneratörlü örnek bir güç sistemi için iletim kayıpları ihmal edilmeden KAOA kullanılarak çözülmüştür. Bu problemler için en iyi çözüm değerleri elde edilmeye çalışılmıştır. Elde edilen bu değerler, literatürdeki diğer yaklaşımlar ile karşılaştırıldığında daha iyi sonuçlar elde etmiştir.

(Uz, 2014), bu tezde ızgara sistemleri, en uygun profil boyutlarına ve minimum ağırlığa sahip olduğu için ateşböceği algoritması yöntemiyle tasarlanmıştır. Tasarım algoritması LRFD-AISC (Load and Resistance Factor Design-American Steel Construction Institute) (Yük ve Direnç Faktörü Tasarımı Amerikan Çelik Konstrüksiyon Enstitüsü) tasarım sınırlayıcılarının sağlanmış olması şartıyla, ızgara sisteminin enine ve uzunlamasına kirişleri için LRFD-AISC'de 272 farklı W-profilinden uygun profili seçer. Oluşturulan her bir tasarımın ağırlığı, onu oluşturan tüm profillerin birim ağırlıkları dikkate alınarak hesaplanır. Tüm iterasyonlardan sonra, minimuma indirilen sistem ağırlığı, algoritma tarafından kaydedilir ve nihai sonuç olarak kabul edilir.

(Arora ve Singh, 2013), geliştirdikleri algoritmada rastgele oluşturulmuş çözümler ateşböcekleri olarak kabul edilmiştir ve objektif parlaklık işlevi üzerindeki performanslarına bağlı çalıştırılmıştır. Algoritma, parametre seçimi kılavuzlarının türetildiği 5 standart karşılaştırma fonksiyonu kullanılarak performans ve başarı oranı

temelinde analiz edilmiştir. Alandaki az sayıda ateşböceklerinde ve daha düşük α ve γ değerlerinde daha iyi sonuçlar elde etmiştir.

(Dekhici ve ark., 2012), güç dağıtımını optimize etmek için ateşböceği algoritmasını sağlamayı amaçlamışlar ve değerlendirme için parçacık sürü optimizasyonu ile karşılaştırmışlardır. Uygulama bir IEEE-14 ve iki termal tesis ağında yapılmıştır. Parçacık sürüsü optimizasyonu ile yapılan karşılaştırmaya göre, ateşböceği algoritmasının bir saniyeden daha kısa sürede en iyi maliyete ulaştığı belirterek etkinliği gösterilmiştir.

(Yang ve ark., 2012), yaptıkları çalışmada ateşböceği algoritması kullanarak Ekonomik Sevkiyat (ES) problemlerine uyarlamışlardır. Güç jeneratörlerinin pek çok doğrusal olmayan özelliği ve üretim kısıtlamaları, yasaklı çalışma alanları, rampa hızı sınırları, iletim kaybı ve doğrusal olmayan maliyet fonksiyonları gibi operasyonel kısıtlamaları, pratik çalışma için tasarlamışlar. Önerilen yöntemin etkinliğini ve uygulanabilirliğini göstermek için, konveks olmayan çözüm alanlarına sahip olan ve en son yayınlanan ES çözüm yöntemlerinin bazılarıyla karşılaştırılan dört ES test sistemini incelemişlerdir. Bu çalışmanın sonuçları, önerilen ateşböcekleri diğer yöntemlerle belirlenenlerden daha ekonomik yükler bulabildiğini göstermiştir.

(Gandomi ve ark., 2011), yaptıkları çalışmada karışık sürekli/ayrık yapısal optimizasyon problemlerini çözmek için ateşböceği algoritmasını kullanmıştır. Literatürden elde edilen 6 (welded beam design, pressure vessel design, helical compression spring design, reinforced concrete beam designs, stepped cantilever beam design ve car) klasik yapısal optimizasyon problemi üzerine yapılan çalışmanın sonuçları, önerilen algoritmanın geçerliliğini doğrulamaktadır. Optimizasyon sonuçları ateşböceğinin PSO, GA, SA ve HS gibi diğer metaheuristik algoritmalardan daha verimli olduğunu göstermektedir.

(Khadwilard ve ark., 2011), bu makalede JSSP'nin çözümü için ateşböceği algoritması önerilmiştir. Önerilen algoritmanın parametre ayarı ile ilgili araştırma yapmışlar ve ateşböceği parametresini çeşitli parametre ayarlarıyla karşılaştırmışlardır. OR-Kütüphanesinden alınan 5 kıyaslama JSSP veri seti üzerinde testler yapılmıştır. Optimize edilmiş parametre ayarları kullanılarak ateşböceği performansı üzerine deney sonuçlarının analizi gerçekleştirilmiştir. Deneme analizinden elde edilen uygun parametre ayarlı ateşböceği, parametre ayarlarını kabul etmeden ateşböceğinden daha iyi olanı daha iyi bir program üretmiştir.

(Apostolopoulos ve Vlachos, 2010), bu makalede ateşböceği algoritmasını ekonomik emisyon yükü dağıtımının çok amaçlı minimizasyon problemine uygulama için önermişler ve test etmişler. Tek bir kirletici nitrojen oksit olan NOx'in azaltılması üzerineydi; iletim kayıpları, güç dengesinin eşitlik kısıtlaması ve eşitsizlik üreteci kapasitesi kısıtlamaları da dikkate almışlardır. Bu önerilen yöntemin uygulanmasının sonuçları, belirli optimizasyon problemini çözmek ve gerçek Pareto optimal çözümlerini bulmak için ateşböceği algoritmasının etkinliğini açıkça göstermiştir. Algoritma, diğer stokastik doğadan esinlenerek elde edilen sonuçlarla karşılaştırmışlar iyi sonuçlar elde etmişlerdir.

(Karakoyun ve Babalik, 2015), bu makalede Karışık Kurbağa Sıçrama Algoritması (KKSA) (Shuffled Frog Leaping Algorithm (SFLA)), sosyal kurbağaların davranışlarına dayanarak modellenmiştir. KKSA, benchmark problemleri üzerine bölümsel veri kümelenmesi için kullanılmıştır. Kullanılan veri setleri UCI makine öğrenme veri ambarından alınmıştır. KKSA'nın performansı Yapay Arı Kolonisi (ABC) algoritması, Parçacık Sürü Optimizasyonu (PSO) algoritması ve dokuz sınıflandırma algoritması ile karşılaştırılmıştır. Deneysel sonuçlar, KKSA'nın kümelenme problemleri için meta-sezgisel algoritmalara karşı rekabetçi bir alternatif olmuştur.

(Kuila ve Jana, 2014), bu makalede enerji verimli kümeleme ve yönlendirme, kablosuz algılayıcı ağlarının kullanım ömrünü uzatmak için geniş çapta araştırılan iki iyi bilinen optimizasyon problemidir. Bu problemlerin doğrusal/doğrusal olmayan programlama formülasyonları ve bunu takiben, parçacık sürüsü optimizasyonu (PSO) önermişlerdir. Önerilen algoritmalar kapsamlı olarak denenmiş ve sonuçlar ağ ömrü, enerji tüketimi, ölü algılayıcı düğümleri ve toplam veri paketlerinin baz istasyonuna iletimi açısından üstünlüklerini göstermek için mevcut algoritmalar ile karşılaştırmışlar. Deneysel sonuçlar, önerilen algoritmaların ağ ömrü, aktif olmayan sensör düğüm sayısı ve toplam veri paketleri iletimi açısından mevcut algoritmalardan daha iyi performans göstermiştir.

(Kuo ve ark., 2014), bu makalede yeni bir otomatik kümelenme algoritması önerilerek arı kolonisi optimizasyonu (bee colony optimization) ile otomatik çekirdek kümelenmesi (automatic kernel clustering) (AKC-BCO) algoritmaları birleştirilmiştir. AKC-BCO, uygun sayıda kümeyi belirler ve kümeleri düzeltmek için veri noktaları atar. Bu, küme yeteneğini artıran çekirdek işleviyle gerçekleştirilir. Bu yöntem birkaç karşılaştırma veri seti kullanılarak doğrulanmıştır. Sonuç, mevcut birkaç otomatik kümeleme metodu ile karşılaştırılmaktadır. Deney sonuçları önerilen AKC-BCO'nun

diğerlerinden daha kararlı ve doğru olduğunu göstermektedir. Ayrıca, önerilen yöntem, gerçek bir dünya tıbbi sorununa da uygulanır.

(Hatamlou, 2013), bu makalede kara delik (KD) fenomeninden esinlenen yeni bir sezgisel algoritma önermiştir. Diğer nüfus tabanlı algoritmalara benzer şekilde, kara delik algoritması (KD), bir başlangıç çözüm adayı optimizasyon problemine ve onlar için hesaplanan objektif bir işleve başlar. Kara delik algoritmasının her yinelemesinde, en iyi aday kara delik olarak seçilir ve daha sonra yıldız denilen etraftaki diğer adayları çekmeye başlar. Bir yıldız kara deliğe çok yaklaşırsa, kara delik tarafından yutulacak ve sonsuza dek yok olacaktır. Böyle bir durumda, yeni bir yıldız (aday çözüm) rastgele oluşturulur ve arama alanına yerleştirilir ve yeni bir arama başlatır. Kara delik algoritmasının performansını değerlendirmek için, NP-zor problem olan kümelenme problemini çözmek için uygulanır. Deneysel sonuçlar, önerilen kara delik algoritmasının, birkaç kıyaslama veri kümesi için diğer geleneksel sezgisel algoritmalardan daha üstün olduğunu göstermiştir.

(Rana ve ark., 2013), bu makalede sınır kısıtlama stratejisi ile Sınır Kısıtlı Uyarlamalı Parçacık Swam Optimizasyonu (SKUPSO) algoritması olarak adlandırılan yeni geliştirilmiş bir algoritma önermişlerdir. Önerilen SKUPSO algoritması dokuz veri kümesinde test edilmiş ve sonuçları PSO, NM-PSO, K-PSO ve K-ortalama kümeleme algoritmaları ile karşılaştırılmıştır. Önerilen algoritmanın daha doğru sonuçlar verdiği ve yakınsama hızının diğer algoritmalarla karşılaştırıldığında daha hızlı olduğu bulunmuştur.

(Kuo ve ark., 2012), bu makalede parçacık sürüsü optimizasyonu (PSO) ve genetik algoritmaya (GA) dayalı yeni bir dinamik kümeleme (DKPG) yaklaşımı önermişler. Önerilen DKPG algoritması, önceden belirlenmiş sayıda küme olmaksızın verileri inceleyerek verileri otomatik olarak kümeleyebilir. Dört karşılaştırmalı veri setinin hesaplamalı sonuçları, DKPG algoritmasının, ikili-PSO (DKPSO) ve dinamik kümeleme GA (DKGA) algoritmalarına dayalı dinamik kümeleme yaklaşımına dayalı dinamik kümeleme yaklaşımından daha iyi bir geçerlilik ve kararlılığa sahip olduğunu göstermiştir. Kümelenme sonuçları, aynı materyalleri paylaşan ürünleri kümelere ayırmak için kullanılabilir.

(Agustı ve ark., 2012), bu makalede kümelenme problemleri için yeni bir gruplama genetik algoritması önermişler. Algoritmanın performansının iyileştirilmesi için algoritmaya dahil edilen yerel arama ve ada modelinin tanımını tam olarak tanımlamışlar. Önerilen gruplama genetik algoritmasını, kamu deneylerinden elde edilen

sentetik ve gerçek verilerdeki çeşitli deneylerde test etmişler ve sonuçlarını, K-ortalama ve DBSCAN algoritmaları gibi klasik kümelenme yaklaşımları ile karşılaştırarak, önerilen gruplamanın iyiliğini doğrulayan mükemmel sonuçlar elde etmişler.

(Yan ve ark., 2012), bu makalede veri kümeleme için bir hibrit yapay arı koloni (HABC) algoritması kullanmışlar. HABC'nin mekanizması, Genetik Algoritma (GA) 'nın çapraz operatörünü ABC'ye tanıtarak arılar arasındaki bilgi alışverişini (sosyal öğrenme) geliştirmektir. On benchmark fonksiyonu üzerinde yapılan bir testle, önerilen HABC algoritmasının, kanonik ABC ve bazı diğer karşılaştırma algoritmalarına göre anlamlı bir iyileşme sağladığı kanıtlanmıştır. HABC algoritması daha sonra veri kümeleme için kullanmışlar. UCI makine öğrenme ambarından seçilen altı gerçek veri kümesi kullanmışlar. Sonuçlar, HABC algoritmasının diğer algoritmalardan daha iyi sonuçlar aldığını ve veri kümelenmesi için rekabetçi bir yaklaşım olduğunu göstermiştir.

(Niknam ve ark., 2011), bu makalede, verimli hibrit evrimsel optimizasyon algoritması olarak Modify Imperialist Competitive Algorithm (MICA) ve K-ortalama (K) adlandırılan K-MICA birleştirilmesine dayanan bir optimizasyon algoritması önermişler. Yeni hibrit K-ICA algoritması, çeşitli veri kümelerinde test edilmiştir ve performansı, MICA, ACO, PSO, Simüle Annealing (SA), Genetik Algoritma (GA), Tabu Search (TS), Honey Bee Mating Optimizasyonu ile karşılaştırılmıştır. Simülasyon sonuçları, önerilen evrimsel optimizasyon algoritmasının, veri kümelemesini ele almak için sağlam ve uygun olduğunu göstermiştir.

(Tsai ve Kao, 2011), bu makalede partikül sürüsü optimizasyonuna (PSO) dayanan yeni bir algoritma olan seçici rejenerasyon partikül sürtünme optimizasyonu (SRPSO) önermişlerdir. İki yeni özellik, dengesiz parametre ayarı ve parçacık rejenerasyonu işlemi içerir. Bu algoritma, performans değerlendirmesi için veri kümeleme problemlerine uygulanmıştır ve K-ortalama kümeleme yönteminin bir hibrit algoritması (KSRPSO) ve SRPSO geliştirilmiştir. Yapılan sayısal deneylerde, SRPSO ve KSRPSO, orijinal PSO algoritması, K-ortalama ve diğer çalışmalar tarafından önerilen diğer yöntemlerle karşılaştırılmıştır. Sonuçlar, SRPSO ve KSRPSO'nun veri kümeleme problemleri için etkili, doğru ve sağlam yöntemler olduğunu göstermiştir.

# 3. MATERYAL VE YÖNTEM

Literatürde bu güne kadar araştırmacılar tarafında geliştirilen birden fazla kümeleme algoritmaları bulunmaktadır. Bu tezde ateşböceği optimizasyon algoritması tabanlı kümeleme yaklaşımı kullanılmıştır. Ateşböceği optimizasyon algoritması kümelemede küme merkezlerinin güncellenmesi için kullanılmıştır.

## 3.1. Test (Benchmark) problemleri

Test problemleri optimizasyon algoritmasının performans ölçümü için kullanılmıştır. Bu çalışmada kullanılan 12 adet test problemleri (Balance, Breast Cancer Wisconsin Diagnostic, Breast Cancer Wisconsin orijinal, Credit, Dermatology, Pima Diabetes, E. Coli, Glass, Heart, Iris, Newthyroid, Wine) University of California, Irvine (UCI) web sitesinde yer alan ve açık erişime sahip veri tabanından alınmıştır (Veritabanı, 2018). Aşağıda sırayla bu veri setleri hakkında kısa bilgi verilmiştir.

*Balance veri seti:* Bu veri seti, psikolojik deney sonuçlarını modellemek için oluşturulmuştur. Her bir örnek, denge skalası ucunu sağa, ipucu sola veya dengelenmiş olarak sınıflandırılır. Öznitelikler sol ağırlık, sol mesafe, sağ ağırlık ve sağ uzaklıktır. Sınıfı bulmanın doğru yolu (sol-mesafe, sol-ağırlık) ve (sağ-mesafe, sağ-ağırlık) daha büyüktür. Eşit ise dengeli. Bu veri seti 4 özelliğe, 3 sınıfa ve 625 örneğe sahiptir (Veritabanı, 2018).

*Breast Cancer Wisconsin Diagnostic veri seti:* Veri setinde yer alan özellikler bir göğüs kütlesinin ince iğne aspiratının sayısallaştırılmış görüntüsünden hesaplanır. Görüntüde mevcut olan hücre çekirdeğinin özelliklerini tarif ederler. Bu veri seti 32 özelliğe, 2 sınıfa ve 569 örneğe sahiptir (Veritabanı, 2018).

*Breast Cancer Wisconsin orijinal veri seti:* Dr. Wolberg klinik vakalarını bildirdiği için numuneler periyodik olarak gelir. Bu nedenle veri tabanı kronolojik verilerin gruplandırılmasını yansıtmaktadır. Bu veri seti 10 özelliğe, 2 sınıfa ve 699 örneğe sahiptir (Veritabanı, 2018).

*Credit veri seti:* Bu veri seti kredi kart uygulamaları ile ilgilidir. Tüm özellik isimleri ve değerleri, verilerin gizliliğini korumak için anlamsız sembollere dönüştürülmüştür. Bu veri seti 51 özelliğe, 2 sınıfa ve 690 örneğe sahiptir (Veritabanı, 2018).

*Dermatology veri seti:* En büyük sınıfa sahip veri kümesidir. Veri setindeki sınıflar dermatolojik hastalıkları (Psoriasis, Seboreik Dermatit, Liken Planus, Pityriasis Rosea Kronik Dermatit ve Pityriasis Rubra Pilaris) içerir. Bu veri seti 33 özelliğe, 6 sınıfa ve 366 örneğe sahiptir (Veritabanı, 2018).

*Pima Diabetes veri seti:* Diyabet hasta kayıtları iki kaynaktan elde edilmiştir. Bunlar; otomatik bir elektronik kayıt cihazı ve kağıt kayıtlarıdır. Otomatik cihaz, zaman damgası olayları için dahili bir saate sahipken, kağıt kayıtları sadece "mantıksal zaman" yuvaları (kahvaltı, öğle yemeği, akşam yemeği, yatma zamanı) sağlamıştır. Kâğıt kayıtları için kahvaltıda sabit zamanlar (08:00), öğle yemeği (12:00), akşam yemeği (18:00) ve yatma vakti (22:00) olarak sıralanabilir. Böylece kağıt kayıtlar hayali tekdüze kayıt sürelerine sahipken, elektronik kayıtlar daha gerçekçi zaman aralıkları içermektedir. Bu veri seti 8 özelliğe, 2 sınıfa ve 768 örneğe sahiptir (Veritabanı, 2018).

*E. coli veri seti:* Bu veri seti 7 özelliğe, 8 sınıfa ve 336 örneğe sahiptir. Veri setinde yer alan referanslar bu veri setine ve geliştirilmesine öncülük yapmaktadır. Ayrıca, veri kümesinin bu sürümüyle kural tabanlı bir uzman sistem tarafından sınıflandırma için sonuçlar da verirler. Referans: "gram negatif bakterilerde protein lokalizasyon sitelerinin tahmin edilmesi için uzman sistem", Kenta Nakai ve Minoru Kanehisa, PROTEİNLER: Yapı, Fonksiyon ve Genetik, Referans: "ökaryotik hücrelerde protein lokalizasyon sitelerini tahmin etmek için bir bilgi tabanı", Kenta Nakai ve Minoru Kanehisa, Genomik (Veritabanı, 2018).

*Glass veri seti:* Bu veri seti cam tiplerini tanımlar. Bunlar; düz işlenmiş bina pencereleri, şamandırasız işlenmiş bina pencereleri, araç pencereleri, konteynerler, masa gereçleri olarak sıralanabilir. Cam türlerinin sınıflandırılması çalışması, kriminolojik araştırmalarla motive edilmiştir. Bu veri seti 9 özelliğe, 7 sınıfa ve 214 örneğe sahiptir (Veritabanı, 2018).

*Heart disease veri seti:* Bu veri seti hastadaki kalp hastalığının varlığına işaret eder ve 0'dan (yok) 4'e kadar tamsayılardır. Bu veri seti 75 özelliğe, 2 sınıfa ve 303 örneğe sahiptir (Veritabanı, 2018).

*İris veri seti:* Bu, örüntü tanıma literatüründe bulunan belki de en iyi bilinen veri tabanıdır. İris veri kümesi her biri 50 örnekten oluşan 3 sınıf içerir, burada her sınıf bir çeşit iris bitkisine atıfta kümesi bulunur. İris veri setinde 4 (sepal uzunluğu, sepal genişliği, petal uzunluğu ve petal genişliği) özellikten oluşmaktadır (Veritabanı, 2018).

*Newthyroid veri seti:* Bu veri seti 3 sınıf içerir (tiroid fonksiyonu olarak: fonksiyon, normal fonksiyon veya fonksiyon altında). Veri seti 5 özellik, 3 sınıf ve 215 örneğe sahiptir (Veritabanı, 2018).

*Wine veri seti:* Bu veriler İtalya'daki aynı bölgede yetişen fakat üç farklı çeşitten türetilen şarapların kimyasal analizlerinin sonuçlarıdır. Analiz üç şarap türünün her birinde bulunan 13 unsurun miktarlarını belirlemektedir. Şarap veri setinde 13 sayısal özellikli 178 örnek vardır (Veritabanı, 2018).

## 3.2. Kümeleme ve Kümeleme Yöntemleri

Kümeleme veri madenciliğinde en sık kullanılan tekniklerden biridir. Verilerin kümelere ayrılması için bir yöntemdir, her kümenin diğer küme verileri ile büyük benzerlik ve yüksek benzerlik olmayan verilere sahip olması gerekir (Rana ve ark., 2013). Kümeleme, veri nesnelerini önceden bilinmeyen gruplara, yani kümeler halinde organize etme işlemidir. Aynı kümedeki veri nesneleri birbirine benzer ve diğer kümelerdeki veri nesnelerine benzemez. Kümeleme denetlenmeyen bir süreçtir, kümeleme yöntemleri sadece veri noktalarının benzerliklerini kullanarak verilerin içindeki ilişkileri bulur. Kümeleme, verilerin analizinde çok yararlı olan verilerin bölümlerini bulmak için kullanılır. Verilerin özelliklerini tanımlamak için bir uzman tarafından etiketlenebilirler veya benzer veri nesnelerini tanımlamak için kullanılır (Akşehirli, 2011).

### 3.2.1. Bölümleme tabanlı kümeleme algoritması

Bölümlemeli (partition based) yöntemde veri setindeki veriler daha önceden belirlenmiş kritere göre ayrılır. Bölümlemeli yönteminde her veri bölünen bağımsız bir kümede yer alır (Demirkan, 2014). Kümeleme işlemi, her nesnenin mutlaka bir bölüm içinde yer almasına ve her bölümün en az bir nesneden oluşma esaslarına dayanır (Tekbir, 2009). K-ortalama algoritması en iyi bilinen ve yaygın olarak kullanılan bir kümeleme algoritması ve bölümleme tekniğidir. İlk kez J. MacQueen tarafından 1967'de tanıtılmıştır (MacQueen, 1967). Bu yöntem yıllardır bilimsel ve endüstriyel uygulamalarda en yaygın kullanılan kümeleme algoritması haline gelmiştir. Algoritmanın çalışmasında öncelikle sabit bir küme sayısına ihtiyacı vardır. Küme sayısı *k* ile gösterilir ve elemanlarının birbirine olan yakınlığına göre oluşturulacak grupların sayısını ifade eder. Kümeleme işlemi, verilerin en yakın veya benzer küme merkezlerine yerleştirerek

gerçekleştirilir. Çalışma yönteminde genellikle, Denklem 3.1 de verilen öklid bağlantısı temel alınarak kümeleme yapılmaktadır. Eğer küme sayısı belirtilmemişse deneme yolu ile en uygun sayı bulunur veya bu değer algoritmaya dışardan verilir. *K* adet rastgele küme merkezi belirlenir veya ilk eleman merkez olabilir. Elemanların merkezlerine yakınlıkları hesaplanarak, yakın oldukları merkezlere göre (Denklem 3.2) kümeleme yapılır. Oluşan kümelerin ortalamaları hesaplanarak yeni küme merkezleri belirlenir. Bu işlem kümelenecek eleman kalmayıncaya kadar devam eder (Çerezci, 2015 ). K-ortalama algoritmasında veri setindeki *x* değerinin tüm küme merkezi $c_i$ ile olan uzaklıkları aşağıdaki Öklid uzaklık denklemine göre hesaplanır, *x* değeri en yakın küme merkezine atanır.

$$d(c_i, x) = \sqrt{(c_i - x)^2} \tag{3.1}$$

$$\min d(c_i, x)^2 \tag{3.2}$$

*x, veri nesnesi,*

*$c_i$, i küme merkezidir,*

*d, uzaklık.*

*K-ortalama algoritmanın adımları*

Adım 1: Kullanıcı tarafından küme sayısı (*K*) verilir.

Adım 2: Küme sayısı kadar rastgele küme merkezi üretilir.

Adım 3: Her objenin küme merkezlerine olan uzaklıkları öklid denklemine göre hesaplanır.

Adım 4: Nesne en yakın olan kümeye atanır.

Adım 5: İterasyon sayısı sona erdiyse en son küme işlemi kabul edilir, değilse aşağıdaki denkleme göre küme merkezleri güncellenir ve adım 3'e gider.

$$\text{Güncelleme: } \left(\frac{\text{kümedeki eleman değerlerin toplamı}}{\text{kümedeki eleman sayısı}}\right) \tag{3.3}$$

### 3.2.2. Hiyerarşik tabanlı kümeleme algoritması

Hiyerarşik algoritmalar toplayıcı ve bölücü olmaktadır. Bölücü algoritmalar tüm küme ile başlayıp veri setini daha küçük kümelerle böler ve yapılan işlem büyük kümeleri küçük kümelere parçalama işlemidir (Avcı, 2006; Güler, 2006). Toplayıcı algoritmalar her elemanı ayrı bir küme alarak başlar ve daha büyük kümeler olacak şekilde birleştirir

veya küçük kümeleri büyük bir kümede birleştirme işlemidir (Avcı, 2006) (Güler, 2006). Hiyerarşik kümelerde veriler kümelere tek bir adımda bölünemezler. Onun yerine tüm nesneleri içeren tek bir kümeden, her biri tek nesne içeren n kümeye doğru bir seri bölümleme uygulanır (Avcı, 2006). Hiyerarşik algoritmanın adımları aşağıda verilmiştir.

*Hiyerarşik algoritmanın adımları*

Adım 1. Her nesneler için değer oluşturulur ve her nesneyi bir küme olarak ata. (Bu durumda N tane nesne için N adet küme oluşur.) N x N bir matris oluşturulur.

Adım 2. En yakın olan iki nesneyi grupla. Matrisi kullanarak nesneler arası uzaklık değerlerini hesapla.

Adım 3. N=N-1 boyutunda yeni matrisi oluştur. N değeri 1 oluncaya kadar Adım 2. ye geri dön (Uzun, 2016).

### 3.2.3. Yoğunluk tabanlı kümeleme algoritması

Yoğunluğa dayalı yöntemde, belirli bir yoğunluğa yakın noktaların oluşturduğu belirli bir veri tabanında kümeler bulunmaktadır. Bu tür kümeler herhangi bir şekilde olabilir. Veri seti içindeki elemanların yoğunluğuna bağlı olarak kümelenebilir. Veri tabanındaki herhangi bir kümeye ait olmayan bir dizi veri noktasının gürültü olduğu düşünülür (Tekbir, 2009). Bu yöntemde kümelemedeki ana fikir, veri setindeki nesneleri yakınlıklarına göre yoğunluk şartlarına dayanan kümelere ayırmaktır. Veri seti içindeki bireylerin yoğun olduğu bölgeler ve bu veri setini düşük yoğunluktaki bölgelere bölmeye çalışmaktadır. Bu kümelemeye göre noktaları sonlu sayıda kümeye bölmek için, yoğunluk ve sınır gibi kavramların bilinmesi gerekir. Bu kavramların noktaların en yakın komşularıyla ilgili olmaktadır. Yoğunluğa dayalı kümeleme algoritmaları keyfi şekilli kümeleri ortaya çıkarmaktadır (Güler, 2006). Bu tür kümeleri bulmak diğer birçok kümeleme algoritmalarından farklı olarak ‘‘Nokta yoğunluğu (point density)’’ kavramı ile veriler ilişkilendirilmektedir. Algoritmanın çalışma mantığı 4 farklı parametre almaktadır (Kayım, 2015 ).

*Points:* kümeleme yaptığımız veri setini ifade eder.
*Distance:* Ne kadar uzaklık için kümeleme yapılacağı bu parametre ile belirlenir.

*Epsilon:* genellikle çok küçük pozitif bir sayıyı ifade eden parametredir. Bu değer dataPoint p'nin epsilon komşuluk değerlerinin belirlenmesi için kullanılır. P noktası epsilon değerinden küçük ya da değerine eşit uzaklıktaki noktalar ile komşudur. Bu yüzden ‘‘epsilon komşuluğu’’ şeklinde ifade edebilir.
*MinPoints:* eğer bir nokta bir kümeye ait ise fakat çekirdek noktalardan biri değil ise bu durumda sınır (border) noktalardan biri olmaktadır (Kayım, 2015 ).

Algoritma işlem mantığını aşamalı olarak incelememiz gerekiyorsa:

Adım 1. Rastgele daha önce işlem yapılmamış bir nokta seçilir.
Adım 2. Seçilen bu rastgele noktanın epsilon uzaklığı içerisindeki komşuları bulunur.
Adım 3. Eğer komşu sayısı *minPoints* sayısına eşit ya da büyük ise bir küme oluşturulur, *minPoints* sayısından küçük ise bu nokta gürültü olarak işaretlenir.
Adım 4. Bir nokta bir kümeye dahil edildi ise bu durumda tüm epsilon komşuları da kümeye dahil edilir. Tabii ki bu eklenenler için de geçerlidir. Bu işlem eklenecek nokta kalmayıncaya kadar devam ettirilir ve böylece kümenin tamamı oluşturulmuş olur.
Adım 5. Yeni üzerinden geçilmemiş rastgele bir nokta seçilir ve döngü tekrarlanır (Kayım, 2015 ).

### 3.2.4. Bulanık mantık kümeleme algoritması

Bulanık mantık kavramı, 1965 yılında California'da Berkeley Üniversitesinde Prof. A. Lotfi Zadeh tarafından ilk kez tanıtıldı. Bu tarihten sonra, giderek gün gittikçe artan bulanık mantık, belirsizlikler ve belirsizliklerle çalışmak için oluşturulmuş katı bir matematiksel düzen olarak tanımlanabilir. Bilindiği gibi, istatistiksel ve olasılık teorisi belirsizlik yerine kesin olarak incelenir, ancak kişinin yaşadığı çevre belirsizlik ile doludur. Bu nedenle, insanlığın sonuç oluşturma kabiliyetini anlamak için belirsizlikler ile çalışmak gereklidir. Bulanık mantık ve matematik arasındaki temel fark, bilinen bir anlamda sadece aşırı matematik değerlerine izin vermesidir. Karmaşık sistemleri klasik matematiksel yöntemler ile modellemek ve denetlemek zordur, çünkü veriler tam olmalıdır. Bulanık yöntemi, kümeler birbirinden belirgin bir şekilde ayrılamamışsa veya üyeliklerinde bazı elemanlar küme üyeliğinde kararsızsa uygun bir yöntem olmaktadır. Bulanık mantıktan anlaşıldığı, mantık kuralları esnek ve bulanık bir şekilde uygulanmaktadır. Klasik mantıktan bilindiği gibi ‘‘doğru ve yanlış’’ ya da ‘‘1’’ ve ‘‘0’’ vardır, oysa bulanık mantıkta, önermeler ve ifadeler kabul edilir. Gerçek hayatta kesinlikle yanlış değil. Geçek hayatta önermeler genellikle kısmen doğru veya

muhtemelen doğru kabul edilir. Bulanık mantığa zaten ihtiyaç duyulmaktadır, çünkü klasik mantık gerçek dünya problemleri için yeterli değildir. Bulanık mantık sisteminde bir ifade tamamen yanlışsa, klasik mantıkta olduğu gibi 0 olur, ya da tamamen doğruysa 1 olur (ancak bulanık mantık uygulamalarının çoğu, bir ifadenin 0 veya 1'e izin vermez veya yalnızca çok özel durumlarda). Bunların haricinde, tüm ifadeler gerçek değerler 1'den küçük ve 0'dan büyüktür (Güler, 2006).

# 4. OPTİMİZASYON

Optimizasyon, bir problemde belirli şartlar altında en iyi çözümü elde etme işlemidir (Akyol ve Alataş, 2012). Optimizasyon, bir fonksiyonun maksimumunu veya minimumunu veren koşulları bulma işlemi olarak tanımlanabilir. En basit anlamda, bir optimizasyon bir minimizasyon veya maksimizasyon problemin çözümüdür. Araştırmacılar tarafından geliştirilen PSO, ABC ve GA gibi bir çok optimizasyon algoritması bulunmaktadır ve bu optimizasyon algoritmaları farklı problem çözümlerinde başarılı bir şekilde kullanılmıştır. Bu çalışmada Ateşböceği (Firefly) optimizasyon algoritması kümeleme işlemi için kullanılmıştır. Ateşböceği algoritmasının çalışma prensibi detaylı olarak aşağıda açıklanmıştır.

## 4.1. Ateşböceği Optimizasyon Algoritması

Ateşböceği Algoritması (Firefly Algorithm (FA)), tropikal iklim bölgelerinde yaşayan ateşböceklerinin sosyal davranışlarını baz alan bir metasezgisel optimizasyon algoritmasıdır. Cambridge Üniversitesi'nde çalışan araştırmacı Dr. Xin-She Yang tarafından geliştirilmiştir. Ateşböceklerinin ışıklarını aydınlatmasının amacı, diğer ateşböceklerini çekmek için bir sinyal sistemi gibi davranmaktır. Yanıp sönen ışıkların ateş böceklerinin arkadaşlarını bulmakta, avlarını çekmede ve avcılarından korunmasına yardımcı olmaktadır (Yang, 2009). Ateşböceği algoritmasının üç özel idealize edilmiş kuralı vardır bunlar:

(1) Tüm ateşböcekleri tek bir cins olarak kabul edilir daha çekici ve parlak olanlara doğru hareket ederler.

(2) Çekicilik, parlaklık ile orantılıdır, dolayısıyla herhangi iki yanıp sönen ateşböceği için, daha az parlak olanı daha parlak olana doğru hareket eder ve her ikisi de uzaklıkları arttıkça azalır. Belirli bir ateşböceğinden daha parlak bir tane yoksa rastgele hareket eder.

(3) Bir ateşböceğinin parlaklığı veya ışık yoğunluğu, belirli bir problemin amaç fonksiyonunun değeri ile belirlenir. Maksimizasyon problemleri için ışık şiddeti, hedef fonksiyonun değeri ile orantılıdır (Apostolopoulos ve Vlachos, 2010). Şekil 4.1'de ateşböceği algoritmasının pseudo kodu verilmiştir (Yang, 2009).

Amaç Fonksiyonu f(x), $x = (x_1, \dots, x_d)^T$

ateşböceğinin başlangıç popülasyonun oluşturulması $x_i$ $(i = 1,2, \dots, n)$

$x_i$ deki ışık yoğunluğu $I_i$ , $f(x_i)$ ile belirlenir.

Işık emilim katsayısı tanımlanır $\gamma$

While (t<maxjenerasyon).

For i=1:n (tüm ateş böcekleri için)

For j=1:i (tüm ateş böcekleri için) (iç halka)

if ($I_i < I_j$ ), (Ateş böcekleri i den j ye doğru hareket eder).

$$x_i = x_i + \beta_0 e^{-\gamma r^2_{ij}}(x_j - x_i) + \alpha(\text{rand} - 0.5)$$

End if

Çekicilik, mesafeye göre değişir $e^{-\gamma r^2}$

Yeni sonuçlar değerlendirilir ve ışık yoğunluğu güncellenir.

End for j

End for i

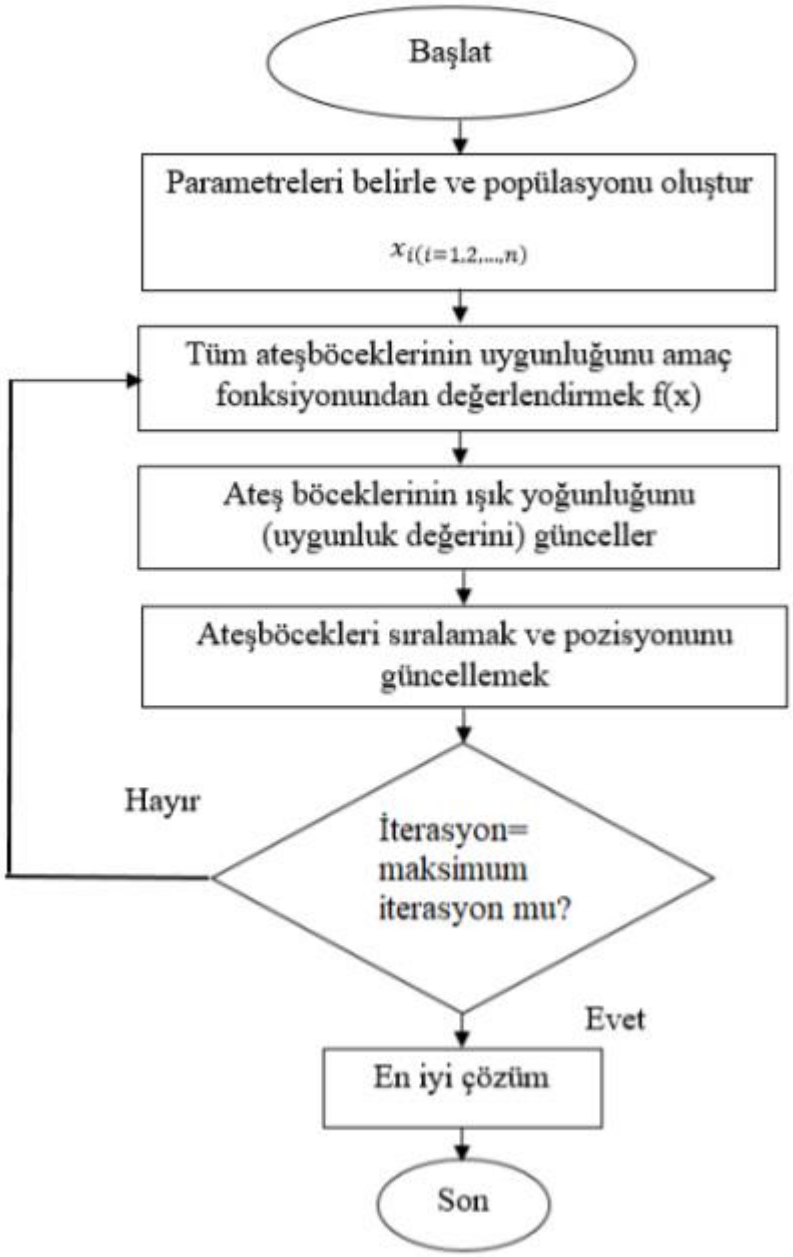
Ateşböcekleri sıraya konulur ve güncel global en iyi bulunur (g*)

End while

Postprocess sonuçları ve görselleştirme

Şekil 4.1. Ateşböceği algoritmasının pseudo kodu

Ateşböceği algoritmasında iki önemli konu vardır: ışık yoğunluğu değişimi ve çekiciliğinin formülasyonu. Ateşböceğinin çekiciliği parlaklığa ve uzaklığa bağlı olarak değişmektedir. Her bir ateşböceğine ait farklı bir çekicilik değeri bulunmaktadır. Şekil 4.2'de ateşböceği optimizasyon algoritmasının akış şeması göstermektedir.




Şekil 4.2. Ateşböceği optimizasyon algoritmasının akış şeması

Çekicilik değeri "β" ile gösterilir, ateşböceğinin çekiciliğinin işlevi Denklem 4.1 de verilmiştir.

$$\beta_{(r)} = \beta_0 \exp(-\gamma r^m), \qquad m \geq 1, \tag{4.1}$$

Burada $\beta_r$, iki ateşböceklerinin arasındaki mesafe, $\beta_0$, bir ateşböceğinin diğer komşu ateşböceğinin aradaki mesafe r=0 olduğundaki başlangıç çekicilik miktarıdır ve $\gamma$ Işığın soğurma katsayısıdır, Işık yoğunluğunun azalmasını kontrol eden bir emilim katsayısıdır (Apostolopoulos ve Vlachos, 2010).

*Mesafe: i* ve *j* iki ateşböceği ve iki boyutlu düzlemde sırasıyla konumları *xi(xi,yi)* ve *xj(xj,yj)* olsun. Aralarındaki mesafe ($r_{ij}$) Öklid bağıntısı hesaplanmaktadır.

$$r_{ij} = \|x_i - x_j\| = \sqrt{(x_i - x_j)^2 - (y_i - y_j)^2} \quad (4.2)$$

*Hareket:* Daha çekici (yani daha parlak) ateşböceği j'ye çekilen ateşböceğinin hareketi aşağıdaki denkleme göre hesaplanmaktadır.

$$x_i = x_i + \beta_0 e^{-\gamma r^2_{ij}}(x_j - x_i) + \alpha\left(\text{rand} - \frac{1}{2}\right) \quad (4.3)$$

α (rand - 1/2), ateşböceğinin rastgele bir hareketidir. α katsayısı, α $\epsilon$ [0-1] ile ilgilenilen problem ile belirlenen bir randomizasyon parametresidir. α ∈ [0,1] aralığında sabit bir değer alan katsayı parametresidir. rand [0,1] arasında rastgele bir değerdir. $\beta_0$ temel cazibe katsayısı olup genellikle $\beta_0 = 1$ olarak değer almaktadır (Yang, 2009).

# 5. ATEŞBÖCEĞİ ALGORİTMASI İLE KÜMELEME

Kümeleme işleminde en iyi küme merkezlerinin bulunması NP-zor problemidir. NP-zor problemlerin çözümünde araştırmacılar birden fazla yöntem önermiştir. Ateşböceği algoritması elde ettiği başarıdan dolayı birçok problem çözümünde kullanılmıştır. Bu çalışmada Rand index uygunluk fonksiyonuna sahip ateşböceği algoritması kümeleme işleminde optimum küme merkezlerini bulmak için kullanılmıştır. Aşağıda tezde kullanılan bu yaklaşım detaylı olarak anlatılmıştır.

## 5.1. Kullanılan Yöntem

Ateşböceği kümeleme algoritmasında kümeleme sonuçlarının değerlendirilmesinde Rand index uygunluk fonksiyonu değerlendirme ölçütü olarak kullanılmıştır. Rand index; N adet nesneden oluşan ve k tane kümeye bölünen bir veri kümesinde, gerçek küme bilgisi bilinen nesneler Beklenen (B) ve kümeleme işleminden sonra nesneler Elde edilen (E) temsil etmektedir. Veri kümesindeki $x_i$ ve $x_j$ nesne çiftinin, B ve E durumlarında birbirine göre 4 farklı durumu vardır. Bunlar:

1. $x_i$ ve $x_j$ nesne çiftinin *E* durumunda aynı kümeye dâhil olması ve *B* durumunda da bu nesne çiftinin aynı kümede olması.
2. $x_i$ ve $x_j$ nesne çiftinin *E* durumunda aynı kümeye dâhil olması ve *B* durumunda da bu nesne çiftinin farklı kümelerde olması.
3. $x_i$ ve $x_j$ nesne çiftinin *E* durumunda farklı kümelere dâhil olması ve *B* durumunda da bu nesne çiftinin aynı kümede olması.
4. $x_i$ ve $x_j$ nesne çiftinin *E* durumunda farklı kümelere dâhil olması ve *B* durumunda da bu nesne çiftinin farklı kümelerde olması.

Veri kümesindeki $x_i$ ve $x_j$ nesne çiftlerinin yukarıdaki 4 durum için elde edilen kümeleme sonucu Rand index değeri Denklem 5.1 deki gibi hesaplanmıştır.

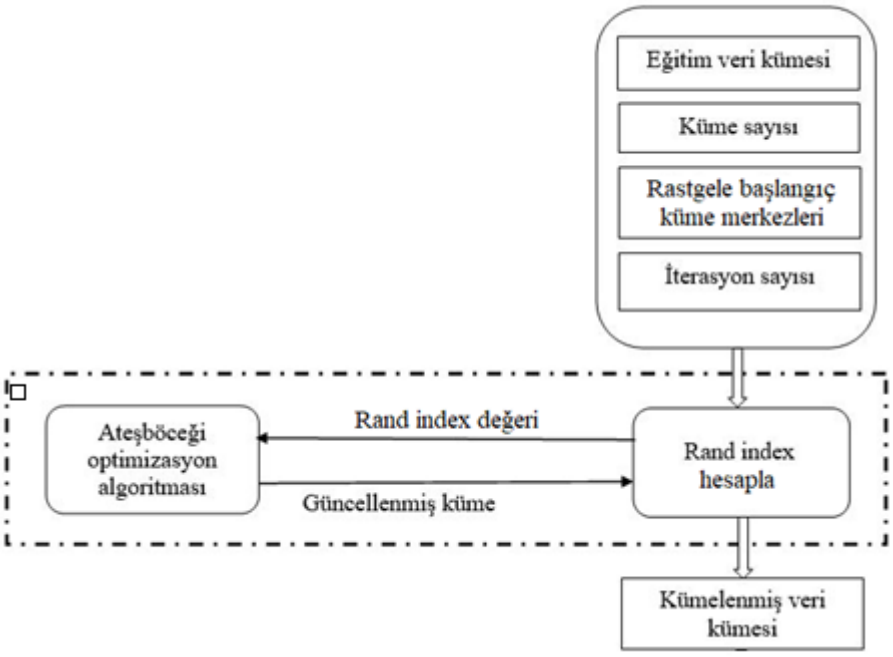
$$\text{Rand index} = \frac{\text{Doğru sayısı}}{\text{Doğru sayısı+yanlış sayısı}} \tag{5.1}$$

Rand index değeri (0,1) aralığında değerler almaktadır. Rand index değeri 1'e yaklaştıkça B ve E durumlarının birbirine benzerliği artmaktadır. Bu da kümelemenin iyi yapıldığını göstermektedir (Karakoyun, 2015).

Ateşböceği algoritması global çözümleri daha geniş bir arama uzayından elde edebilir. Uygulanan ateşböceği algoritması tabanlı kümeleme yönteminin adımları aşağıda verilmiştir.

Adım 1: veri setini oku
Adım 2: ateşböceği algoritmasının parametrelerini ayarla (alfa, beta, gama, ateşböceği sayısı, iterasyon sayısı)
Adım 3: ateşböceği sayısı kadar rastgele başlangıç küme merkezleri üret ve bu küme merkezlerine göre uygunluk fonksiyonu (Rand index) hesapla
Adım 4: iterasyon sayısı kadar küme merkezlerini güncelle
Adım 5: elde edilen çözümleri sırala ve en iyi çözümü al, bu çözüme göre verileri kümele

Şekil 5.1'de önerilen algoritmanın akış diyagramı verilmiştir.

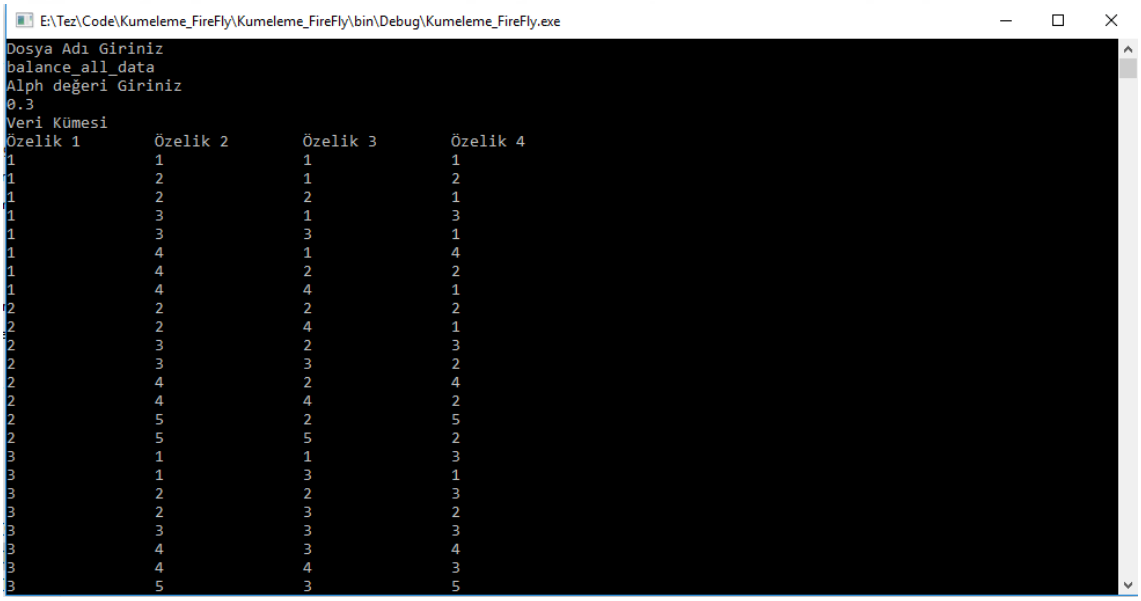



Şekil 5.1. Uygulanan algoritmanın akış diyagramı

Şekil 5.1'de görüldüğü gibi ilk aşamada, veri setinin başlangıç küme merkezleri arama uzayına göre ateşböceği sayısı kadar rastgele atanmaktadır. Atanan rastgele küme

merkezlerine göre veri seti kümelenir, bu kümeleme işleminde uzaklık Öklid denklemi kullanılmaktadır daha sonra her ateşböceği için Rand index uygunluk fonksiyonu hesaplanır. Bir sonraki iterasyonda bu küme merkezlerinin güncellenmesi kısmında ise Ateşböceği algoritması kullanılmaktadır. Kümeleme algoritmasında Rand index fonksiyonu, uygunluk fonksiyonu olarak kullanılmıştır ve her iterasyonda maksimize edilmeye çalışılmıştır.

Uygulanan yöntemin başarısını test etmek amacıyla visual studio C#.Net platformunda bir yazılım geliştirilmiştir. Şekil 5.2'de gösterildiği gibi uygulama yazılımında kümeleme yapılacak dosya ismi (.txt) ve alfa değeri (0 – 1 arasında) girilmekte ve işlem başlatılmaktadır.




Şekil 5.2. Uygulama yazılımının ekran görüntüsü

İşlem bittiğinde ise kümeleme sonuçları metin (*.txt) dosyasına yazdırılmaktadır. Bu dosyada kümeleme başarı yüzdesi ve çalışma süresi (saat:dakika:saniye biçiminde) kaydedilmektedir. Yazdırma işlemi yapılırken her test verisi için alfa değerine göre dosya ismi verilmektedir. Örneğin iris verisi için algoritma çalıştırıldığında alfa değeri 0.1 olarak girilirse dosya adı "iris_all_data1_Sonuçları.txt" olmaktadır.

## 5.2.Uygulama Sonuçları

Tüm testler intel core i5 -3317U işlemcisi, 6 GB RAM ve Windows 10 işletim sistemine sahip olan kişisel bilgisayarda yapılmıştır. Önerilen ateşböceği kümeleme

algoritmasının başarısı sıkça kullanılan 12 adet UCI veri kümesi üzerinde test edilmiş ve önerilen kümeleme algoritmasının sonuçları (Karakoyun, 2015 ) tez çalışmasındaki sonuçları ile karşılaştırılmıştır. Çizelge 5.1'de çalışmada kullanılan veri kümelerinin özelikleri verilmiştir.

Çizelge 5.1. Veri kümelerinin özelikleri

| **Veri seti adı** | **Özellik sayısı** | **Örnek sayısı** | **Sınıf sayısı** |
|---|---|---|---|
| Balance | 4 | 625 | 3 |
| Breast Cancer Wisconsin Diagnostic | 32 | 569 | 2 |
| Breast Cancer Wisconsin Orijinal | 10 | 699 | 2 |
| Credit | 51 | 690 | 2 |
| Dermatology | 33 | 366 | 6 |
| E. Coli | 7 | 336 | 8 |
| Glass | 9 | 214 | 7 |
| Heart Disease | 75 | 303 | 2 |
| İris | 4 | 150 | 3 |
| Pima Diabetes | 8 | 768 | 2 |
| Newthyroid | 5 | 215 | 3 |
| Wine | 13 | 178 | 3 |

Uygulanan kümeleme algoritmasında Rand index fonksiyonu uygunluk fonksiyonu olarak kullanılmıştır ve bu fonksiyon her iterasyonda maksimize edilmeye hedeflenmiştir. Algoritmanın ilk aşamasında küme merkezleri veri setlerinin en küçük ve en büyük değerlerine göre atanmıştır daha sonra bu arama uzayında en iyi küme merkezleri aranmıştır.

Kümeleme algoritmasında ateşböceği algoritmasının başarısını ve performansını etkileyen alfa vektörü bulunmaktadır. Genelde literatürde alfa değeri 0 ile 1 arasında alınmaktadır. Bu tez çalışmasında da alfa değerleri 0 ile 1 arasında 0.1 aralıklı olarak artırılarak kümeleme sonuçları kaydedilmiştir. Bu kümeleme sonuçlarından en iyi olanı alınmıştır. Önerilen ateşböceği kümeleme algoritmasında sırasıyla beta, gama, iterasyon sayısı ve ateşböceği sayısı değerleri 1, 1, 1000 ve 10-20 olarak ayarlanmıştır. Sonuçların kararlılığını belirleyebilmek için 20 kez çalıştırılmıştır ve elde edilen değerlerin ortalaması ele alınmıştır. Çizelge 5.2'de Rand index sonuçları verilmiştir.

Çizelge 5.2. Önerilen ateşböceği kümeleme algoritması Alfa değerlerinde elde edilen rand index sonuçları

| Veri kümesi | Alfa değerleri | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 0,1 | 0,2 | 0,3 | 0,4 | 0,5 | 0,6 | 0,7 | 0,8 | 0,9 |
| Balance | **0,858** | 0,803 | 0,797 | 0,784 | 0,798 | 0,779 | 0,802 | 0,806 | 0,808 |
| Breast Cancer Wisconsin Diagnostic | **0,728** | **0,728** | **0,728** | **0,728** | **0,728** | **0,728** | **0,728** | **0,728** | **0,728** |
| Breast Cancer Wisconsin Orijinal | 0,900 | 0,926 | 0,935 | 0,920 | 0,926 | **0,946** | 0,935 | 0,931 | 0,930 |
| Credit | 0,667 | 0,667 | 0,667 | 0,671 | **0,690** | 0,684 | 0,667 | 0,667 | 0,688 |
| Dermatology | 0,940 | 0,940 | 0,952 | 0,963 | 0,970 | 0,964 | **0,975** | 0,963 | 0,960 |
| E. Coli | 0,813 | 0,771 | 0,780 | **0,850** | 0,804 | 0,777 | 0,780 | 0,777 | 0,765 |
| Glass | **0,769** | **0,769** | **0,769** | **0,769** | **0,769** | **0,769** | **0,769** | **0,769** | **0,769** |
| Heart Disease | 0,660 | 0,668 | 0,673 | 0,680 | 0,680 | 0,687 | **0,715** | 0,693 | 0,695 |
| İris | 0,913 | 0,920 | 0,933 | 0,910 | 0,940 | **0,967** | 0,940 | 0,953 | **0,967** |
| Pima Diabetes | 0,664 | 0,669 | 0,681 | 0,673 | 0,712 | **0,734** | 0,685 | 0,732 | 0,699 |
| Newthyroid | 0,744 | 0,805 | 0,795 | 0,828 | 0,847 | 0,860 | 0,865 | **0,879** | 0,870 |
| Wine | 0,930 | 0,940 | 0,942 | 0,969 | 0,959 | **0,976** | 0,960 | 0,960 | 0,960 |

Çizelge 5.2 incelendiğinde Balance veri kümesi 4 özellik, 3 sınıf ve 625 örnekten oluşmaktadır. Alfa değeri 0.1 iken önerilen ateşböceği kümeleme algoritması 0.858 değerinde en iyi kümeleme sonucunu verdiği görülmektedir. Breast Cancer Wisconsin Diagnostic veri kümesi 32 özelliğe, 569 sınıfa ve 2 örneğe sahiptir alfa değeri 0.1'den 0.9'a kadar önerilen ateşböceği kümeleme algoritması aynı değeri 0.728, Breast Cancer Wisconsin Orijinal veri kümesi 10 özelliğe, 699 sınıfa ve 2 örneğe sahiptir alfa değeri 0.6 da önerilen ateşböceği kümeleme algoritması 0.946, Credit veri kümesi 51 özelliğe, 690 sınıfa ve 2 örneğe sahiptir alfa değeri 0.5 te önerilen ateşböceği kümeleme algoritması 0.690, Dermatology veri kümesi 33 özelliğe, 366 sınıfa ve 6 örneğe sahiptir alfa değeri 0.7 de önerilen ateşböceği kümeleme algoritması 0.975, E. Coli veri kümesi 7 özelliğe, 336 sınıfa ve 8 örneğe sahiptir alfa değeri 0.4 de önerilen ateşböceği kümeleme algoritması 0.850, Glass veri kümesi 9 özelliğe, 214 sınıfa ve 7 örneğe sahiptir alfa değeri 0.1'den 0.9'a kadar önerilen ateşböceği kümeleme algoritması 0.769, Heart Diseas veri kümesi 75 özelliğe, 303 sınıfa ve 2 örneğe sahiptir alfa değeri 0.7 da önerilen ateşböceği kümeleme algoritması 0.715, Iris veri kümesi 4 özelliğe, 150 sınıfa ve 3 örneğe sahiptir alfa değeri 0.6 ve 0.9 da önerilen ateşböceği kümeleme algoritması 0.967, Pima Diabetes veri kümesi 8 özelliğe, 768 sınıfa ve 2 örneğe sahiptir alfa değeri 0.6 da önerilen ateşböceği kümeleme algoritması 0.734, Newthyroid veri kümesi 5 özelliğe, 215 sınıfa ve 3 örneğe sahiptir alfa değeri 0.8 de önerilen ateşböceği kümeleme algoritması 0.879, Wine veri kümesi 13 özelliğe, 178 sınıfa ve 3 örneğe sahiptir alfa değeri 0.6 da önerilen ateşböceği kümeleme algoritması 0.976 en iyi kümeleme sonucunu vermiştir.

Tüm bu sonuçlardan en iyi kümeleme sonuçları (Karakoyun, 2015 ) tez çalışmasındaki Karışık Kurbağa Sıçrama Algoritması (KKSA) ile yapılan sonuçlarla karşılaştırılmıştır ve karşılaştırma sonuçları Çizelge 5.3'te verilmiştir.

Çizelge 5.3. Ateşböceği kümeleme algoritması ile KKSA sonuçlarının karşılaştırılması

| Veri kümesi | Karışık Kurbağa Sıçrama Algoritması | Ateşböceği algoritması |
|---|---|---|
| Balance | 0,762 | **0,858** |
| Breast Cancer Wisconsin Diagnostic | **0,888** | 0,728 |
| Breast Cancer Wisconsin Orijinal | 0,926 | **0,946** |
| Credit | **0,773** | 0,690 |
| Dermatology | **0,981** | 0,975 |
| E. Coli | **0,900** | 0,850 |
| Glass | 0,743 | **0,769** |
| Heart Disease | 0,683 | **0,715** |
| İris | 0,930 | **0,967** |
| Pima Diabetes | 0,600 | **0,734** |
| Newthyroid | **0,930** | 0,879 |
| Wine | 0,962 | **0,976** |

Karşılaştırma sonuçlarına bakıldığında, önerilen ateşböceği kümeleme algoritması sırasıyla Balance, Breast Cancer Wisconsin Orijinal, Glass, Heart Disease, İris, Pima Diabetes ve Wine veri kümelerinde 0.858, 0.946, 0.769, 0.715, 0.967, 0.734 ve 0.976 değerleri ile karışık kurbağa sıçrama (KKSA) kümeleme algoritmasından daha iyi kümeleme sonuçları elde etmiştir. Breast Cancer Wisconsin Diagnostic, Credit, Dermatology, E. Coli ve Newthyroid veri kümelerinde ise 0,888, 0.773, 0.981, 0.900 ve 0.930 değerlerinde KKSA algoritması önerilen ateşböceği kümeleme algoritmasından daha iyi kümeleme başarısı vermiştir. Veri setlerinin gerek özellik değerleri, gerek örnek sayıları ve gerekse sınıf sayılarına baktığımızda farklılıklara sahip olduğu görülmektedir. Alınan sonuçlar incelendiğinde daha iyi performans gösteren yöntemin veri seti parametrelerinden (özellik değeri, sınıf sayısı vb) bağımsız olarak bu sonucu ürettiğini görmekteyiz. Dolayısıyla bu iki yöntemin veri parametrelerinden etkilenmediği sonucunu çıkarabiliriz.

Bununla birlikte, önerilen ateşböceği kümeleme algoritmasından elde edilen sonuçlar (Karakoyun ve Babalik, 2015) yayınındaki KKSA ve PSO optimizasyon temeli kümeleme algoritmaları ile karşılaştırılmıştır. Karşılaştırma, sınıflandırma hata yüzdesi (SHY) (Classification Error Percentage (CEP)) denklemine göre (Denklem 5.2) yapılmaktadır. Çizelge 5.4'te uygulanan ve KKSA ve PSO kümeleme algoritmalarının karşılaştırmalı sonuçları verilmiştir.

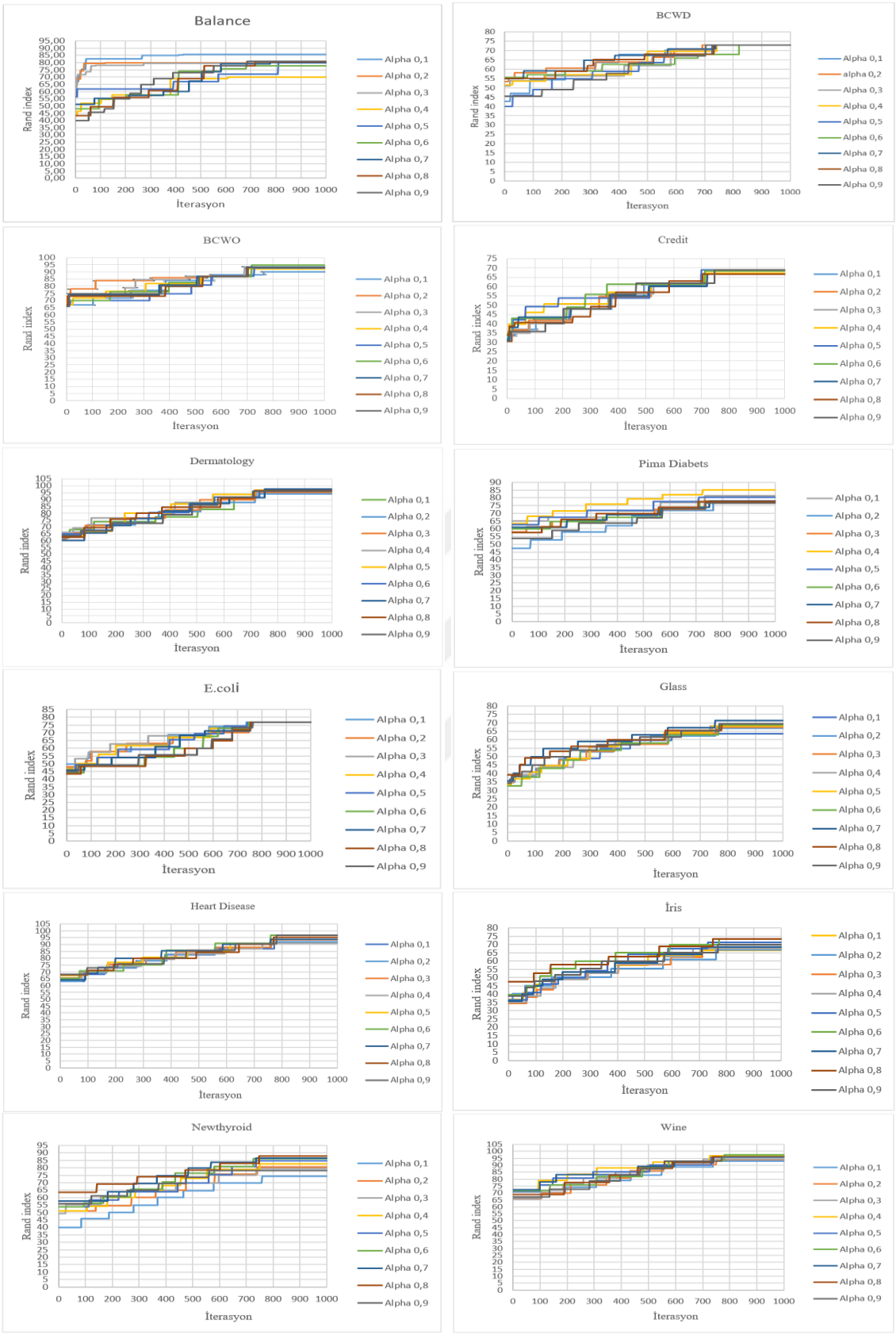
$$\text{SHY} = \frac{\text{Yanlış kümelenen örnek sayısı}}{\text{Toplam örenk sayısı}} \times 100 \quad (5.2)$$

Çizelge 5.4. Önerilen ateşböceği kümeleme algoritması, KKSA ve PSO optimizasyon temeli kümeleme algoritmalarının SHY sonuçları.

| Data Set | Ateşböceği Alg. | KKSA | PSO |
|---|---|---|---|
| Balance | **14,24** | 28,33 | 25,47 |
| Breast Cancer Wisconsin Diagnostic | **4,241** | 6,42 | 5,8 |
| Breast Cancer Wisconsin Orijinal | 5,4 | 4,01 | **2,87** |
| Credit | 16,015 | **13,77** | 22,96 |
| Dermatology | **2,53** | 3,93 | 5,76 |
| E. Coli | 14,98 | **14,15** | 14,63 |
| Glass | **23,11** | 43,35 | 39,05 |
| Heart Disease | 28,5 | 20,92 | **17,46** |
| Iris | 3,334 | 7,22 | **2,63** |
| Pima Diabetes | 26,56 | 28,81 | **22,5** |
| Thyroid | **4,094** | 5,08 | 5,55 |
| Wine | 2,36 | 2,88 | **2,22** |

Çizelge 5.4'de baktığımızda Ateşböceği yönteminde Balance, Breast Cancer Wisconsin Diagnostic, Dermatology, Glass ve Thyroid veri setlerinde sırasıyla 14.24, 4.241, 2.53, 23.11 ve 4.094 en iyi değerleri elde etmiştir. KKSA yönteminde Credit ve E. Coli veri setlerinde sırasıyla 13.77 ve 14.15 en iyi değerleri elde etmiştir. PSO yönteminde Breast Cancer Wisconsin Orijinal, Heart Disease, Iris, Pima Diabetes ve Wine veri setlerinde sırasıyla 2.87, 17.46, 2.63, 22.5 ve 2.22 en iyi değerleri elde etmiştir. Tüm veri kümeleri göz önüne alındığında Ateşböceği algoritması ile PSO algoritmasının 5 adet veri kümesinde en iyi sonucu ürettiği görülmektedir. Önceki karşılaştırma sonuçlarına ilişkin yaptığımız yorumda da belirtildiği üzere kümeleme performansları veri setlerine ait parametrelerden (özellik değeri ve sınıf sayısı) bağımsız olduğu görülmektedir.

Tüm veri setleri için önerilen ateşböceği algoritmasından alfa değerini 0.1 artırarak 0.9'a kadar elde edilen kümeleme sonuçlarının yakınsama grafikleri Şekil 5.3'te gösterilmiştir.




Şekil 5.3. Tüm veri setleri için yakınsama eğrileri

Çalışmada kümeleme işlemine ait işlem süreleri (saat:dak:san) Çizelge 5.5'te verilmiştir.

Çizelge 5.5. Önerilen ateşböceği kümeleme algoritması çalışma süreleri (saat:dak:san)

| Veri kümesi | Alfa değerleri | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 0,1 | 0,2 | 0,3 | 0,4 | 0,5 | 0,6 | 0,7 | 0,8 | 0,9 |
| Balance | **00:14:00** | 00:13:19 | 00:13:38 | 00:13:33 | 00:13:38 | 00:13:48 | **00:13:09** | 00:13:30 | 00:13:12 |
| Cancer | 01:58:04 | 01:57:43 | 01:58:05 | **01:57:14** | 01:57:37 | **02:01:36** | 01:57:25 | 01:58:03 | 01:57:24 |
| Cancer-orij. | **00:33:24** | 00:28:19 | 00:27:14 | 00:27:44 | 00:27:30 | 00:27:22 | 00:27:23 | **00:27:09** | 00:27:16 |
| Credit | **00:46:38** | 00:47:18 | 00:50:19 | 00:48:49 | 00:47:11 | 00:47:37 | 00:49:33 | **00:50:35** | 00:49:21 |
| Dermatology | 01:44:48 | 01:45:24 | **01:44:02** | 1:44:07 | 01:45:19 | **01:50:37** | 01:43:22 | 01:45:33 | 01:43:46 |
| E. Coli | 05:40:11 | 05:40:13 | **05:38:60** | 05:39:14 | 05:40:33 | 05:45:15 | 05:53:02 | **05:54:26** | 05:40:28 |
| Glass | 00:11:29 | 00:11:28 | 00:11:27 | 00:11:24 | 00:11:15 | **00:11:13** | 00:11:35 | 00:11:33 | **00:11:60** |
| Heart Diseas | **00:27:05** | 00:23:19 | 00:24:06 | 00:23:33 | 00:26:26 | 00:23:09 | 00:22:37 | **00:22:18** | 00:23:23 |
| Iris | **00:03:51** | **00:03:13** | 00:03:22 | 00:03:14 | 00:03:36 | 00:03:25 | 00:03:15 | 00:03:32 | 00:03:17 |
| Pima Diabetes | 00:27:02 | 00:33:03 | 00:32:37 | 00:26:52 | **00:39:24** | 00:27:00 | 00:29:10 | 00:28:23 | **00:26:26** |
| Thyroid | 00:05:43 | **00:06:10** | 00:05:39 | **00:05:24** | 00:05:37 | 00:05:39 | 00:05:47 | 00:05:28 | 00:05:52 |
| Wine | 00:12:48 | 00:12:34 | **00:13:25** | **00:12:22** | 00:13:23 | 00:12:34 | 00:12:40 | 00:13:04 | 00:12:44 |

Çizelge 5.5'e bakıldığında;

- Balance veri kümesinde en az ve en çok zaman alan kümeleme süresi sırasıyla 00:13:09 ve 00:14:00 kümelenmiştir.
- Cancer veri kümesinde en az ve en çok zaman alan kümeleme süresi sırasıyla 01:57:14 ve 02:01:36 kümelenmiştir.
- Cancer Orijinal veri kümesinde en az ve en çok zaman alan kümeleme süresi sırasıyla 00:27:09 ve 00:33:24 kümelenmiştir.
- Credit veri kümesinde en az ve en çok zaman alan kümeleme süresi sırasıyla 00:46:38 ve 00:50:35 kümelenmiştir.
- Dermatology veri kümesinde en az ve en çok zaman alan kümeleme süresi sırasıyla 01:43:22 ve 01:50:37 kümelenmiştir.
- E. Coli veri kümesinde en az ve en çok zaman alan kümeleme süresi sırasıyla 05:38:60 ve 05:54:26 kümelenmiştir.
- Glass veri kümesinde en az ve en çok zaman alan kümeleme süresi sırasıyla 00:11:13 ve 00:11:60 kümelenmiştir.
- Heart Disease veri kümesinde en az ve en çok zaman alan kümeleme süresi sırasıyla 00:22:18 ve 00:27:05 kümelenmiştir.
- Pima Diabetes veri kümesinde en az ve en çok zaman alan kümeleme süresi sırasıyla 00:26:26 ve 00:33:03 kümelenmiştir.

- İris veri kümesinde en az ve en çok zaman alan kümeleme süresi sırasıyla 00:03:13 ve 00:03:51 kümelenmiştir.
- Newthyroid veri kümesinde en az ve en çok zaman alan kümeleme süresi sırasıyla 00:05:24 ve 00:06:10 kümelenmiştir.
- Wine veri kümesinde en az ve en çok zaman alan kümeleme süresi sırasıyla 00:12:22 ve 00:13:25 kümelenmiştir.

İşlem süreleri en düşük ve en yüksek süre açısından değerlendirildiğinde görülmektedir ki; alfa açısına bağlı olarak birbirine yakın olduğu ancak oransal bir değişim olmadığı, veri seti büyüklüğüne ve karmaşıklığına göre (örnek adedi, özellik adedi ve sınıf sayısı) değişim gösterdiği sonucuna varılmıştır.

Uygulanan ateşböceği kümeleme algoritmasında, alfa değerleri 0 ile 1 arasında olmak üzere ateşböceği sayısı 10 olarak ayarlanmış ve kümeleme sonuçları alınmıştır. Ateşböceği sayısının performansa etkisini görmek amacıyla önceki uygulamada alınan sonuçlara göre en iyi alfa değerleri alınarak 20 adet ateşböceği sayısı için işlemler tekrar edilmiş ve yeni kümeleme sonuçları kaydedilmiştir. Çizelge 5.6 da bu işlem sonucunda alınan yeni Rand index sonuçları ve çalışma süreleri verilmiştir.

Çizelge 5.6. 10 ve 20 adet ateşböceği kullanılarak alınan rand index kümeleme sonuçları ve çalışma süreleri

| Veri kümesi | Alfa | 10 Ateşböceği | Süre (saat:dak:san) | 20 Ateşböceği | Süre (saat:dak:san) |
|---|---|---|---|---|---|
| Balance | 0.1 | 0,858 | 00:14:00 | **0,872** | 00:19:00 |
| Breast Cancer Wisconsin Diag. | 0.1 | 0,728 | 01:58:04 | **0,728** | 02:06:00 |
| Breast Cancer Wisconsin Orig. | 0.6 | 0,946 | 00:27:22 | **0,948** | 00:43:03 |
| Credit | 0.5 | 0,690 | 00:47:11 | **0,697** | 01:24:00 |
| Dermatology | 0.7 | **0,975** | 01:43:22 | 0,968 | 02:36:08 |
| E. Coli | 0.4 | 0,850 | 05:39:14 | **0,850** | 06:38:00 |
| Glass | 0.1 | 0,769 | 00:11:29 | **0,785** | 00:18:07 |
| Heart Disease | 0.7 | 0,715 | 00:22:37 | **0,742** | 00:36:53 |
| Iris | 0.6 | 0,967 | 00:03:25 | **0,967** | 00:40:06 |
| Pima Diabetes | 0.6 | 0,734 | 00:27:00 | **0,741** | 00:40:58 |
| Newthyroid | 0.8 | 0,879 | 00:05:28 | **0,879** | 00:08:00 |
| Wine | 0.6 | 0,976 | 00:12:34 | **0,976** | 00:20:00 |

Çizelge 5.6'dan da görüldüğü üzere ateşböceği sayısının 10 dan 20 ye çıkarılması kümeleme algoritmasının performansında artış sağlamıştır. Bununla birlikte ateşböceği sayısının artırılması işlem süresini de artırmıştır.

# 6. SONUÇ

Günümüzde birçok araştırmacı verilerin en doğru biçimde kümelenmesi için çalışmalar yapmakta ve değişik yöntemler geliştirmektedir. Ateşböceği algoritması da son yıllarda üzerinde sıklıkla çalışılan optimizasyon ve kümeleme algoritmalarındandır. Bu çalışmada ateşböceği optimizasyon tabanlı bir kümeleme uygulaması yapılmıştır. Önerilen kümeleme yönteminde Rand index uygunluk fonksiyonuna sahip ve küme merkezlerinin güncellenmesi için ateşböceği optimizasyon algoritması kullanılmıştır.

Önerilen yöntemin performansını değerlendirmek için literatür üzerinde sıkça çalışılan UCI veri ambarından alınan veri setleri (Balance, Breast Cancer Wisconsin Diagnostic, Breast Cancer Wisconsin Original, Credit, Dermatology, Pima Diabetes, E. Coli, Glass, Heart Disease, İris, Newhyroid ve Wine) kullanılmıştır. Önerilen kümeleme algoritmasında Rand index fonksiyonu uygunluk fonksiyonu olarak kullanılmıştır ve her iterasyonda bu fonksiyonun maksimize edilmesi hedeflenmiştir. Ateşböceği algoritmasının başarısını ve performansını etkileyen alfa değeri 0 ile 1 arasında 0,1 aralıklarla artırılarak kümeleme sonuçları bulunmuştur.

Önerilen kümeleme algoritmasının test sonuçları, aynı veri setlerini Karışık Kurbağa Sıçrama Algoritması (KKSA) ve PSO ile kümeleyen çalışmadaki (Karakoyun ve Babalik, 2015) (Karakoyun, 2015 ) sonuçlar ile karşılaştırılmıştır.

10 adet ateşböceği kullanılarak yapılan karşılaştırma sonuçlarına göre; önerilen ateşböceği kümeleme algoritması sırasıyla Balance, Breast Cancer Wisconsin Orijinal, Glass, Heart Disease, İris, Pima Diabetes ve Wine veri kümelerinde 0.858, 0.946, 0.769, 0.715, 0.967, 0.734 ve 0.976 değerleri ile KKSA kümeleme algoritmasından daha iyi kümeleme sonuçları elde etmiştir. Breast Cancer Wisconsin Diagnostic, Credit, Dermatology, E. Coli ve Newthyroid veri kümelerinde ise 0,888, 0.773, 0.981, 0.900 ve 0.930 değerlerinde KKSA algoritması önerilen ateşböceği kümeleme algoritmasından daha iyi kümeleme başarısı vermiştir. Veri setlerinin gerek özellik değerleri, gerek örnek sayıları ve gerekse sınıf sayılarına baktığımızda farklılıklara sahip olduğu görülmektedir. Alınan sonuçlar incelendiğinde daha iyi performans gösteren yöntemin veri seti parametrelerinden (özellik değeri, sınıf sayısı vb) bağımsız olarak bu sonucu ürettiğini görmekteyiz. Dolayısıyla bu iki yöntemin veri parametrelerinden etkilenmediği sonucunu çıkarabiliriz.

Bununla birlikte diğer karşılaştırmada KKSA ve PSO optimizasyon temelli kümeleme algoritmaları ile 10 adet ateşböceği içeren kümeleme algoritması

karşılaştırılmıştır. Bu karşılaştırma sınıflandırma hata yüzdesi SHY denklemine göre yapılmıştır. SHY ne göre yapılan karşılaştırmada ise ateşböceği yönteminde Balance, Breast Cancer Wisconsin Diagnostic, Dermatology, Glass ve Thyroid veri setlerinde sırasıyla 14.24, 4.241, 2.53, 23.11 ve 4.094 en iyi değerleri elde etmiştir. KKSA yönteminde Credit ve E. Coli veri setlerinde sırasıyla 13.77 ve 14.15 en iyi değerleri elde etmiştir. PSO yönteminde Breast Cancer Wisconsin Orijinal, Heart Disease, Iris, Pima Diabetes ve Wine veri setlerinde sırasıyla 2.87, 17.46, 2.63, 22.5 ve 2.22 en iyi değerleri elde etmiştir. Tüm veri kümeleri göz önüne alındığında ateşböceği algoritması ile PSO algoritmasının 5 adet veri kümesinde en iyi sonucu ürettiği görülmektedir.

Ateşböceği kümeleme algoritmasında kullanılan ateşböceği sayılarının kümeleme performansında olan etkisini görmek amacıyla 20 adet ateşböceği için işlemler tekrarlanmıştır. İlk çalışmada elde edilen en iyi kümeleme sonuçlarının elde edildiği alfa değerleri kullanılarak yapılan yeni denemede görülmüştür ki ateşböceği sayısının artırılması kümeleme performansında iyileşme sağlamaktadır ancak işlem sürelerinde artışa sebep olmaktadır.

Önerilen optimizasyon temelli kümeleme algoritması global arama yeteneğine sahip olduğundan dolayı ve veri setinin sahip olduğu arama uzayının dışına çıkmadan tüm en iyi küme merkezlerini denetlemektedir, birçok veri setinde standart kümeleme algoritmalarından daha iyi kümeleme başarısı elde etmiştir. Bu karşılaştırma sonuçlarına göre önerilen kümeleme yöntemi farklı kümeleme problemlerine uygulanabilir.

**KAYNAKLAR**


A. J. Umbarkar, U. T. B., and P. D. Seth, 2017, Performance evaluation of firefly algorithm with variation in sorting for non-linear benchmark problems, *Conference Proceedings 1836, 020032 (2017); doi: 10.1063/1.4981972.*

Aderinwale, T. W., 2017, Integrative clustering approaches for cancer subtype discovery, *Master of Science Thesis, Antalya Bilim University, 59s.*

Agustı, L., Salcedo-Sanz, S., Jiménez-Fernández, S., Carro-Calvo, L., Del Ser, J. ve Portilla-Figueras, J. A., 2012, A new grouping genetic algorithm for clustering problems, *Expert Systems with Applications*, 39 (10), 9695-9703.

Akşehirli, M. E., 2011, Newcluster ensemble algorithmwith automatic cluster number and newpruning technique for fast detection of neighbors on binary data, *Master of Science Thesis, Bahçeşehir Üniversitesi, İstanbul, 73s*.

Akyol, S. ve Alataş, B., 2012, Güncel Sürü Zekası Optamizasyon Algoritmaları, *Nevşehir Bilim ve Teknoloji Dergisi*, 1 (1).

Amin, M. A. M., 2016, Kablosuz Algılayıcı Ağlarda Kümeleme Algoritmaları ile Enerji Verimliliğinin Arttırılması İçin Alternatif bir Yöntem Geliştirme, *Doktora Tezi, Gazi Üniversitesi Bilişim Enstitüsü, 90s.*

Apostolopoulos, T. ve Vlachos, A., 2010, Application of the firefly algorithm for solving the economic emissions load dispatch problem, *International journal of combinatorics*, 2011.

Arora, S. ve Singh, S., 2013, The firefly optimization algorithm: convergence analysis and parameter selection, *International Journal of Computer Applications*, 69 (3).

Avcı, U., 2006, Bulanık kümeleme algoritmalarının karşılaştırmalı analizi ve bilgisayar uygulamaları, *Ege Üniversitesi*.

Çerezci, F., 2015 K-Means Algoritması ile Elektromanyetik Kirlilik Analizi, *Yüksek Lisans Tezi, Sakarya Üniversitesi Fen Bilimleri Enstitüsü, 38s.*

Dekhici, L., Borne, P. ve Khaled, B., 2012, Firefly algorithm for economic power dispatching with pollutants emission, *Informatica Economică vol. 16, no 2/2012*, 16 (2), 45-57.

Demirkan, M., 2014, Kategorik ve Kategorik Olmayan Verilerden Oluşan Veri Setleri için K-ortalama Tabanlı bir Yaklaşım, *Yüksek lisans tezi, Beykent Üniversitesi, Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği, İstanbul, 1-74s.*

Erdil, E., 2011, Aggregating Advantages of a Set of Clusterings into a Final Clustering Using Object, Wise Similarity Graph, *Yüksek Lisans Tezi, Bahçeşehir Üniversitesi, 63.*

Uzun, E., 2016, Hiyerarşik kümeleme, https://www.e-adys.com/makine_ögrenmesi/hiyerarşik-kümeleme-hierarchical-clustering-ödev-benzerlikleri-üzerinden-kopya-gruplarını-bulma/

Gandomi, A. H., Yang, X.-S. ve Alavi, A. H., 2011, Mixed variable structural optimization using firefly algorithm, *Computers & Structures*, 89 (23-24), 2325-2336.

Gulati, H. ve Singh, P., 2015, Clustering techniques in data mining: A comparison, *Computing for Sustainable Global Development (INDIACom), 2015 2nd International Conference on*, 410-415.

Güler, N., 2006, Bulanık Kümeleme Analizi ve Bulanık Modellemeye Uygulamaları, *Yayınlanmamış Yüksek Lisans Tezi, Muğla Üniversitesi Fen Bilimleri Enstitüsü, Muğla, 157s*.

Karakoyun, M. ve Babalik, A., 2015, Data Clustering with Shuffled Leaping Frog Algorithm (SFLA) for Classification, *2015 Int'l Conference on Intelligent Computing, Electronics Systems and Information Technology (ICESIT-15)*.

Karakoyun, M., 2015 Kurbağa sıçrama algoritmasının kümeleme problemlerine uygulanması, *Yüksek lisans tezi, Selçuk Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği, Konya, 1-92.*

Kayım, F., 2015 K-MEANS ile DBSCAN algoritması'nın paralelleştirmesi ve hadoop üzerinde büyük veri analizinde kullanılması, performans ve yeterlilik karşilaştırması, *Yüksek Lisans Tezi, Beykent Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği, İsatanbul, 66s.*

Keleş, A., 2017, The clustering of public hospitals for the productivity scorecard application, *M.Sc. Thesis, Ankara Yıldırım Beyazit University, Ankara, 51s.*

Khadwilard, A., Chansombat, S., Thepphakorn, T., Thapatsuwan, P., Chainate, W. ve Pongcharoen, P., 2011, Investigation of firefly algorithm parameter setting for solving job shop scheduling problems, *Operation Research Network in Thailand*, 89-97.

Khalaf, A. Z., 2014, An energy-efficient clustering based communication protocol with dividing the overall network area for wireless sensor networks.

Kuila, P. ve Jana, P. K., 2014, Energy efficient clustering and routing algorithms for wireless sensor networks: Particle swarm optimization approach, *Engineering Applications of Artificial Intelligence*, 33, 127-140.

Kuo, R., Syu, Y., Chen, Z.-Y. ve Tien, F.-C., 2012, Integration of particle swarm optimization and genetic algorithm for dynamic clustering, *Information sciences*, 195, 124-140.

Kuo, R., Huang, Y., Lin, C.-C., Wu, Y.-H. ve Zulvia, F. E., 2014, Automatic kernel clustering with bee colony optimization algorithm, *Information sciences*, 283, 107-122.

Lacroix, R., Salehi, F., Yang, X. ve Wade, K., 1997, Effects of data preprocessing on the performance of artificial neural networks for dairy yield prediction and cow culling classification, *Transactions of the ASAE*, 40 (3), 839-846.

MacQueen, J., 1967, Some methods for classification and analysis of multivariate observations, *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, 281-297.

Nasibov, E., 2012, Çok Boyutlu Veritabanlarında Kümeleme Yöntemleri Üzerine, *Yüksek Lisans Tezi, Ege Üniversitesi Fen Bilimleri Enstitüsü, İzmir, 60s.*

Niknam, T., Fard, E. T., Pourjafarian, N. ve Rousta, A., 2011, An efficient hybrid algorithm based on modified imperialist competitive algorithm and K-means for data clustering, *Engineering Applications of Artificial Intelligence*, 24 (2), 306-317.

Pamuk, N., 2016, Kaotik Ateşböceği Optimizasyon Algoritması Kullanılarak Termik Güç Santralleri Etkisindeki Ekonomik Yük Dağıtım Problemlerinin Çözümü, *Kirklareli University Journal of Engineering and Science*, 2 (1), 38-59.

Rana, S., Jasola, S. ve Kumar, R., 2013, A boundary restricted adaptive particle swarm optimization for data clustering, *International journal of machine learning and cybernetics*, 4 (4), 391-400.

Şişeci, M., 2012, Traverten plaka taşlarda sınıfların kümeleme yöntemleri ile belirlenmesi, *SDÜ Fen Bilimleri Enstitüsü*.

Tekbir, M., 2009, Aykırı değer tespitinde yoğunluk tabanlı kümeleme yöntemleri, *YTÜ Fen Bilimleri Enstitüsü*.

Tsai, C.-Y. ve Kao, I.-W., 2011, Particle swarm optimization with selective particle regeneration for data clustering, *Expert Systems with Applications*, 38 (6), 6565-6576.

Turel, A. ve Can, F., 2011, A new approach to search result clustering and labeling, *Asia Information Retrieval Symposium*, 283-292.

Uz, F. E., 2014, Izgara Sistemlerin Ateşböceği Algoritması Optimizasyon Yöntemi ile Optimum Boyutlandırılması, *Yüksek Lisans Tezi, Celal Bayar Üniversitesi Fen Bilimleri Enstitüsü, Manisa, 84s*.

Veritabanı, U., 2018, UCI Repository Machine Learning Database, https://archive.ics.uci.edu/ml/datasets.html: [12].

Wang, H., Wang, W., Zhou, X., Sun, H., Zhao, J., Yu, X. ve Cui, Z., 2017, Firefly algorithm with neighborhood attraction, *Information sciences*, 382, 374-387.

Yan, X., Zhu, Y., Zou, W. ve Wang, L., 2012, A new approach for data clustering using hybrid artificial bee colony algorithm, *Neurocomputing*, 97, 241-250.

Yang, X.-S., 2009, Firefly algorithms for multimodal optimization, *International symposium on stochastic algorithms*, 169-178.

Yang, X.-S., Hosseini, S. S. S. ve Gandomi, A. H., 2012, Firefly algorithm for solving non-convex economic dispatch problems with valve loading effect, *Applied soft computing*, 12 (3), 1180-1186.

Yıldız, A., 2015, illere göre k-means ve svm kullanılarak sosyal durum analizi, *Süleyman Demirel Üniversitesi fen bilimleri enstitüsü*.

Zhao, Q., 2012, Cluster validity in clustering methods, *Publications of the University of Eastern Finland*.

# ÖZGEÇMİŞ

## KİŞİSEL BİLGİLER


| | | |
|---|---|---|
| **Adı Soyadı** | **:** | **Mina Muayad Abdulazeez ALABD ALRAHMAN** |
| **Uyruğu** | **:** | Irak |
| **Doğum Yeri ve Tarihi** | **:** | Kerkük/ 15.11.1993 |
| **Telefon** | **:** | |
| **Faks** | **:** | |
| **e-mail** | **:** | minoaziz55@gmail.com |


## EĞİTİM

| **Derece** | | **Adı, İlçe, İl** | **Bitirme Yılı** |
|---|---|---|---|
| Lise | : | **Jelal Rıda Efendi, Kerkük, Irak** | **2011** |
| Üniversite | : | **Al-Qalam, Kerkük, Irak** | **2015** |
| Yüksek Lisans | : | **S.Ü. Bilgisayar Mühendisliği A.B.D., Selçuklu Konya** | **2019** |

## UZMANLIK ALANI (Kümeleme, Optimizasyon)

## YABANCI DİLLER (Arabca, ingilizca)

## YAYINLAR

Development of New Clustering Algorithm Based on Firefly Optimization