Journal of the Faculty of Engineering and Architecture of Gazi University 33:2 (2018) 413-423

Mühendislik Mimarlık Fakültesi Dergisi
Journal of The Faculty of Engineering and Architecture of Gazi University

Elektronik / Online ISSN : 1304 - 4915
Basılı / Printed ISSN : 1300 - 1884

New initialization approaches for the k-means and particle swarm optimization based clustering algorithms

Sinem Çınaroğlu, Hasan Bulut*
Department of Computer Engineering, Ege University, İzmir, 35040, Turkey

Highlights:

- Propose new methods for selecting initial cluster center
- Introduce a novel approach for feature selection on multi-dimensional data
- Increase the clustering accuracy for multi-dimensional data

Keywords:

- Clustering
- Particle swarm optimization
- K-means
- Initial centroid selection
- Coreset

Article Info:

Received: 18.03.2016
Accepted: 20.03.2018

DOI:

10.17341/gazimmfd.416350

Acknowledgement:

Correspondence:

Author: Hasan Bulut
e-mail: hasan.bulut@ege.edu.tr
phone: +90 232 311 25 96

Graphical/Tabular Abstract

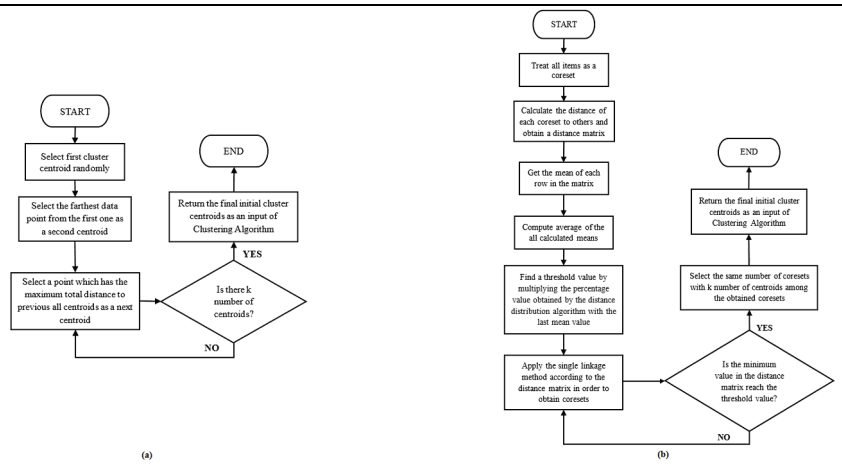

Fig. Proposed methods **a)** Selecting Initial Cluster Centers **b)** Coreset Based Approach

Purpose: It introduces new methods for selecting initial cluster centers of K-means and PSO-based clustering algorithms for multi-dimensional datasets.

Theory and Methods: In the first proposed method Selecting Initial Cluster Centers (tr. Başlangıç Küme Merkezi Seçimi – BMS), first cluster center is selected randomly then the farthest data point is selected as the second cluster center, next a point which has the maximum total distance to previous cluster centers is selected as the third one. This cluster center selection process continues for every cluster center by applying the same selecting strategy as in the third one, until the given cluster numbers is reached. In the other proposed method Coreset Based Approach (tr. Öbek Tabanlı Yaklaşım – ÖTY) firstly, all items are treated as a coreset. A distance matrix is obtained by calculating the distance of each coreset to others. Then, the mean of each row in this matrix and the average of the all calculated means are computed. A threshold value is found by multiplying the percentage value obtained by the distance distribution algorithm with the last mean value. In the next step, the single-linkage method is applied according to the distance matrix in order to obtain coresets. The same process iterates until the minimum value in distance matrix reaches the threshold value. We randomly select the same number of coresets with the given cluster number among the previously obtained coresets. Finally, obtained results are given as the initial cluster center to a clustering algorithm.

Results: The first proposed method, BMS, gives better results in the accuracy index on normalized Breast Cancer, normalized CMC and normalized Heart dataset compared to standard versions and also achieves better results when evaluated from both the accuracy index and the average number of iterations aspects on Heart datasets in which feature selection is already performed. The other proposed method ÖTY results in the highest accuracy index and the best average number of iterations on normalized Yeast, Colon Cancer datasets and normalized Breast Cancer dataset in which feature selection is already performed. However, on the CMC dataset this approach reaches the same success only for the average number of iterations.

Conclusion: Considering the results of the algorithms tested on both normalized datasets and normalized datasets in which feature selection is already performed, the proposed approaches show better clustering performance compared to the results of standard K-means and standard PSO-based clustering algorithms. PSOÖTY algorithm which is one of the proposed approaches has the highest performance for the Colon Cancer microarray dataset. The performance improvement has been shown by means of the experimental results.

Journal of the Faculty of Engineering and Architecture of Gazi University 33:2 (2018) 413-423

Mühendislik Mimarlık Fakültesi Dergisi
Journal of The Faculty of Engineering and Architecture of Gazi University

Elektronik / Online ISSN : 1304 - 4915
Basılı / Printed ISSN : 1300 - 1884

K-ortalamalar ve parçacık sürü optimizasyonu tabanlı kümeleme algoritmaları için yeni ilklendirme yaklaşımları

Sinem Çınaroğlu, Hasan Bulut*
Ege Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, İzmir, 35040, Türkiye,

ÖNEÇIKANLAR

- Başlangıç küme merkezi seçimi için yeni yöntemler önerilmesi
- Çok boyutlu verilerde öznitelik seçimi için yeni bir yaklaşım getirilmesi
- Çok boyutlu verilerin daha yüksek doğrulukta kümelenmesi

Makale Bilgileri
Geliş: 18.03.2016
Kabul: 20.03.2018

DOI:
10.17341/gazimmfd.416350

Anahtar Kelimeler:
Kümeleme,
parçacık sürü optimizasyonu,
k-ortalamalar,
başlangıç merkezi seçimi,
öbek

ÖZET

Mikrodizi teknolojisindeki son gelişmeler sayesinde genlerin farklı seviyelerini eş zamanlı olarak ifade etmek mümkün hale gelmiştir. Genler içindeki gizli bilgilerin temsil edilmesi, genlerin analizini kolaylaştırmakta; ancak gen sayısının fazla olması ve veri setlerindeki yüksek gürültü miktarı gen verilerinin anlaşılmasını zorlaştırmaktadır. Bunun için genlerin belirlenebilmesini kolaylaştırmak amacıyla kümeleme yöntemleri kullanılmaktadır. Mikrodizi verileri çok boyutlu verilere en iyi örneklerdendir. Çok boyutlu verileri kümelemek için çalışma kapsamında standart K-ortalamalar ve Parçacık Sürü Optimizasyonu (PSO) tabanlı kümeleme algoritmaları için başlangıç küme merkezlerinin seçimine yönelik yeni yöntemler önerilmiştir. Ayrıca öbek (coreset) yaklaşımı PSO algoritmasına uyarlanmıştır. Geliştirilen yöntemlerin doğruluğu; literatürde sıkça kullanılan veri setleri üzerinde test edilmiş ve bu yaklaşımlar Colon Cancer mikrodizi veri seti üzerinde çalıştırılmıştır. Baz alınan standart K-ortalamalar ve PSO tabanlı kümeleme yöntemleri ile geliştirilen yaklaşımlar karşılaştırılmış; performansları çözüme ulaşılan ortalama iterasyon sayısı, Rand ve Silhouette indeksleri kullanılarak değerlendirilmiştir. Deneysel çalışmalarda, geliştirilen yaklaşımların öznitelik seçimi yapılmış normalize veri setleri üzerinde başarılı sonuçlar verdiği gözlemlenmiştir.

New initialization approaches for the k-means and particle swarm optimization based clustering algorithms

HIGHLIGHTS

- Propose new methods for selecting initial cluster center
- Introduce a novel approach for feature selection on multi-dimensional data
- Increase the clustering accuracy for multi-dimensional data

Article Info
Received: 18.03.2016
Accepted: 20.03.2018

DOI:
10.17341/gazimmfd.416350

Keywords:
Clustering,
particle swarm optimization,
k-means, initial centroid selection,
coreset

ABSTRACT

Thanks to the recent advances in microarray technology, simultaneously expressing different levels of genes is possible. Although the representation of confidential information in genes simplifies to analyze them; both high number of genes and high amount of noise in the data sets make difficult to identify the gene data. In order to identify genes various clustering methods are generally used. Microarray data is one of the best examples of multidimensional data. In this study, in order to cluster multidimensional data, new methods for selecting initial cluster centers are proposed for the standard K-means and Particle Swarm Optimization (PSO)-based clustering algorithms. Also, coreset approach is adapted for PSO algorithm. The correctness of the developed methods is examined on datasets which are frequently used in the literature, and also these proposed approaches are run on Colon Cancer microarray data set. The performance of the proposed approaches is compared with the standard K-means and PSO-based clustering methods by means of average iteration number, Rand, and Silhouette index metrics. In experimental studies, we observe that proposed methods give superior results on the normalized datasets in which feature selection process is performed.

*Sorumlu Yazar/Corresponding Author: hasan.bulut@ege.edu.tr / Tel: +90 232 311 25 96

1. GİRİŞ (INTRODUCTION)

Kümeleme analizi, bilgi miktarının fazla olduğu veri setlerini gruplandırmak ve verilerdeki önemli bilgileri açığa çıkarmak için kullanılan tekniklerin başında gelmektedir. Grupları önceden belli olmayan veri topluluklarından, ortak özelliklere sahip olan gruplar oluşturmaya yarayan bir yöntemdir. Kümeleme işlemi sonucunda elde edilen küme içinde yer alan veriler arasında maksimum benzerlik ve farklı kümelerde bulunan veriler arasında ise maksimum farklılık oluşması istenir. Biyoinformatik, kümeleme tekniklerinin gelecek vadettiğini gösteren bir uygulama alanıdır [1, 2]. Nitekim mikrodizilerin ve ilgili diğer teknolojilerin yardımı ile ölçülmekte olan gen-ifade (gene-expression) verilerini kümelemenin önemi, geçtiğimiz son yıllarda hızla ve sürekli olarak büyümektedir [3, 4]. Mikrodizi veri setlerindeki yüksek gürültü ve küme sayısının bilinmemesi, eşifadeli genlerin bulunmasını zorlaştırmakta, genleri doğru şekilde kümeleyecek algoritmalara ihtiyaç duyulmaktadır.

Metasezgisel algoritmalardan biri olan PSO; sınıflandırma, kümeleme [5, 6], çizgeleme problemleri, güç ve voltaj kontrolü, çok makinalı güç sistemi kararlı kılıcısı tasarımı [7] gibi bir çok optimizasyon problemi için kullanılmaktadır. Büyük ölçekli kombinasyonel ve doğrusal olmayan problemlerde etkili sonuçlar vermesi, çözüm uzayı tipine ve karar değişken sayısına bağlı olmaması, dönüşümlerinin ve uyarlamalarının kolay olması, hesaplama güçlerinin iyi olması gibi avantajlarından dolayı mühendislik, yönetim bilimi gibi birçok farklı alanda bu algoritmalar kullanılmaktadır [8, 9]. Standart K-ortalamalar ve PSO tabanlı kümeleme algoritmalarının başlangıç küme merkezi seçimi rastgele yapılmaktadır. Bu durum, merkezlerin başlangıçta birbirine yakın olmasına neden olabilmekte ve birbirine yakın olan merkezler, veri setindeki doğal grupların ayrılmasını da mümkün kılabilmektedir. Kümelemedeki grupların birbirinden mümkün olduğunca farklı olma amacı doğrultusunda başlangıç küme merkezlerinin olabildiğince birbirinden uzak seçilmesi, doğruluğu daha yüksek olan kümeler elde edileceğini göstermektedir.

Küme merkezlerinin başlangıç seçimi için çeşitli yöntemler önerilmiştir. Önerilen ilk yöntem veri setinden rastgele başlangıç merkezi seçimi yapan Forgy Yaklaşımı'dır [10]. Bunun yanında, literatürde Kaufman yaklaşımı [11], KKZ yöntemi [12] ve Bradley ve Fayyad yöntemi [13] gibi yöntemler de mevcuttur. Ek olarak Khan ve Ahmad [14] tarafından önerilen Küme Merkezi Başlangıç Algoritması (Cluster Centre Initialization Algorithm-CCIA), bir noktadaki verilerin yoğunluğunu hesaplayan ve daha sonra bu noktaları yoğunluklarına göre sıralayan Yoğunluk-tabanlı Çok Ölçekli Veri Kümeleme (Density-based MultiScale Data Condensation-DBMSDC) yöntemini temel alan bir başlangıç merkezi seçimi yöntemidir. Arai ve Barakbah [15]'ın önermiş olduğu algoritmada K-ortalamalar algoritması ve Hiyerarşik algoritma kullanılmaktadır. Yöntemde, veri setleri üzerinde belirli zamanlarda K-ortalamalar algoritması 10 kez işletilmiş ve buradan elde edilen farklı küme merkezlerinin tamamı kaydedilmiştir. Aynı merkezler farklı hiyerarşik kümeleme (tek bağlantı, merkez bağlantı, tam bağlantı ve ortalama bağlantı) teknikleri kullanılarak yeni küme merkezleri elde edilmiştir. Elde edilen merkezler, K-ortalamalar kümeleme algoritması için en iyi başlangıç küme merkezi olarak kabul edilmiştir. Erişoğlu vd. [16] tarafından yapılan çalışmada, K-ortalamalar kümeleme algoritmasının başlangıç küme merkezini hesaplamak için yeni bir algoritma önerilmiştir. Bu algoritma, en büyük varyasyon katsayısına ve en küçük pozitif korelasyon değerine göre farklı boyutlardaki veri setlerini iki boyuta indirgeme temeline dayanmaktadır. Aggarwal ve Aggarwal [17], başlangıç küme seçimini otomatik oluşturan modifiye bir K-ortalamalar algoritması önermişlerdir. Algoritmada, veri seti negatif öznitelikler içeriyorsa; veri setindeki minimum öznitelik, tüm veri noktalarından çıkartılarak tüm öznitelikler pozitif hale getirilir. Ardından merkeze olan uzaklıkları hesaplanan tüm veriler, bu uzaklıklara göre sıralanır. Sonraki adımda, veri seti k adet parçaya bölünerek, k adet parçanın orta noktaları başlangıç küme merkezi olarak alınır.

Aldahdooh ve Ashour [18], K-ortalamalar algoritmasının başlangıç küme merkezi seçimi için uzaklığa dayalı bir yaklaşım önermiştir. Yaklaşım, rastgele bir nokta seçilerek başlamakta ve bu noktanın gürültü olup olmadığını tespit etmek için seçilen noktaya olan uzaklıklar bulunarak bazı hesaplamalar yapmaktadır. Sonuca göre, gürültü olmadığı tespit edilen noktalar, başlangıç küme merkezi olarak kabul edilmektedir. Qiao ve Lu [19] tarafından önerilen yaklaşımda, başlangıç küme merkezini seçimini hızlandırmak için özellik uzayı, en büyük varyasyon katsayısına ve en küçük pozitif korelasyon değerine göre iki boyuta indirgenir ve olasılıksal yöntemlerle seçilen aday küme merkezleri, yüksek yoğunluklu bölgelere doğru hareket ettirilerek nihai kümeleme sonuçları iyileştirilmiştir. Jothi vd. [20], K-ortalamalar kümeleme algoritmasının başlangıç küme merkezi seçimi için Sınırlandırılmış Özyinelemeli İkiye-Bölümleme kullanarak Deterministik Başlatma (Deterministic Initialization using Constrained Recursive Bi-partitioning) olarak bilinen bir yöntem önermişlerdir. İlk olarak yöntemde olası merkez kümesi, özyinelemeli ikiye bölümleme kullanılarak tanımlanır. Daha sonra K-ortalamalar kümeleme algoritması için gerçek merkezler, olası merkezler üzerinde çizge kümeleme uygulanarak belirlenir. Belirtilenlerin ışığında bu çalışmada, çok boyutlu veri setleri için K-ortalamalar ve PSO tabanlı kümeleme algoritmalarının başlangıç küme merkezi seçimine yeni bir yaklaşım getirilmesi amaçlanmıştır. Bu amaç doğrultusunda 2. bölümde kullanılan algoritmalar hakkında bilgiler verilmiştir. Geliştirilen yaklaşım ayrıntılı olarak 3. bölümde açıklanmış ve bu yaklaşım ile mikrodizi verileri gibi çok boyutlu veri setlerindeki genlerin doğru bir şekilde kümelenmesi hedeflenmiştir. Mikrodizi veri setlerinin boyutları büyük olduğu için geliştirilen yaklaşımların doğruluğu, öncelikle sıklıkla tercih edilen veri

setleri üzerinde test edilmiş, daha sonra mikrodizi veri seti üzerinde çalıştırılmıştır. Yapılan deneyler sonucunda başlangıç küme merkezi seçiminde iyileştirilmeler elde edilmiş ve deneysel sonuçlara 4. bölümde yer verilmiştir. 5. bölümde elde edilen sonuçlar vurgulanmıştır.

2.YÖNTEM (METHOD)

2.1. K-ortalamalar Kümeleme Algoritması
(K-means Clustering Algorithm)

K-ortalamalar algoritması, Mac Queen tarafından 1967 yılında geliştirilen ve çeşitli alanlarda sıkça kullanılan bölümleyici kümeleme yöntemidir [14]. Başlangıçta verilen k küme sayısı kadar rastgele bir paylaşım ile başlar ve nesneler küme merkezlerine olan uzaklıklarına göre kendisine yakın olan kümelere atanırlar. Küme merkezi, kümedeki tüm noktaların tüm koordinatlarının ayrı ayrı aritmetik ortalaması alınarak bulunur [21]. Algoritmanın amacı, oluşturulan kümelerin amaç fonksiyon değerlerini minimize etmektir [6, 22].

k küme sayısı ve *n* eleman sayısı iken veri seti X = (x_1, x_2, …x_n) ve yeni oluşan gruplar G = (g_1, g_2, … g_k) olmak üzere algoritmanın temel adımları aşağıdaki gibidir:

- X = (x_1, x_2, …x_n) veri setinden rastgele C = (c_1, c_2, … c_k) merkezi seç.
- C = (c_1, c_2, … c_k) merkezlerine tüm nesnelerin Öklid uzaklığını hesapla ve kendisine en yakın olan küme merkezine ata.
- C' = (c'_1, c'_2, … c'_k) yeni merkezlerini

$$c'_i = \frac{1}{n_{g_i}} \sum_{x_j \in g_i} x_j \;\; (i = 1,2,\dots,k) \tag{1}$$

  Eş.1 yardımıyla hesapla.
- Eğer maksimum iterasyon sayısına ulaşıldıysa veya küme merkezlerinde bir değişiklik olmadıysa ($c_i = c'_i$) algoritmayı sonlandır. Aksi takdirde $c_i = c'_i$ olarak ata ve 2. adıma geri dön.

K-ortalamalar algoritmasının doğası gereği küme sayısının önceden bilinmemesi ve her seferinde farklı bir küme sayısının rastgele atanmasından dolayı; her çalıştırmada farklı sonuçlar elde edilmektedir. Bununla birlikte bir diğer eksik noktası da küme içi benzerliği sağlamasına rağmen, bu benzerliğin küresel bir benzerlik olduğunu garanti edememektedir. Yani yerel optimuma takılma olasılığı vardır [23].

2.2. Parçacık Sürü Optimizasyonu
(Particle Swarm Optimization-PSO)

PSO, balık ve kuş sürülerinin yiyecek arayışlarından esinlenerek, doğrusal olmayan fonksiyonların optimizasyonu için 1995 yılında James Kennedy ve Russell Eberhart tarafından geliştirilen popülasyon tabanlı bir optimizasyon yöntemidir [24, 25]. Optimum çözüme kararlı bir yakınsama ve kolayca uygulanabilme özelliği olan bu algoritma diğer evrimsel hesaplama ve sürü zekâsı algoritmalarına göre daha az parametre gerektirmektedir. Parçacık (particle) olarak isimlendirilen sürüden oluşan bu algoritmadaki her bir parçacığın optimizasyon problemine çözüm sunan bir konum bilgisini tutan pozisyon vektörü ve yön bilgisini tutan hız vektörü mevcuttur. *d* boyutlu bir veri setinde, *i* parçacığının sahip olduğu konum vektörü $x_i = (x_{i1}, x_{i2}, \dots x_{id})$, hız vektörü $v_i = (v_{i1}, v_{i2}, \dots v_{id})$ şeklindedir ve bu bilgiler her iterasyonda güncellenmektedir. Hız vektörü v_{min} ve v_{max} ile sınırlıdır. Aynı zamanda her parçacığın *pbest (personal best)* olarak isimlendirilen ve nesilden nesile aktarılan en iyi konum vektörü mevcuttur [26]. *gbest (global best)* olarak isimlendirilen vektör ise, parçacık sürüsünün o ana kadar sahip olduğu en iyi konum bilgisini göstermektedir.

İteratif bir algoritma olan PSO algoritmasında, *d* boyutlu bir sette, *i* parçacığının sahip olduğu hız ve konum vektörlerinin her iterasyonda güncellenmesi;

$$v_{id}(t+1) = w.v_{id}(t) + c_1.rand_1. (pbest_{id} - x_{id}) + c_2.rand_2.(gbest_{id} - x_{id}) \tag{2}$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \tag{3}$$

şeklinde olmaktadır. Eş. 2'de *w* atalet değeri, parçacığın hareketsizliğini önlemek için parçacıklara ivme kazandırarak gerekli çeşitliliği sağlamaktadır. c_1 ve c_2 hızlanma sabitleri olup, $rand_1$ ve $rand_2$ ise [0, 1] aralığında rastgele üretilen bir değerdir. PSO algoritması rastgele bir hız ve konum bilgisi alarak çözüm uzayında arama işlemine başlar. Her parçacığın konum bilgisi seçilen uygunluk fonksiyonuna gönderilerek uygunluk değeri elde edilir. Elde edilen uygunluk değeri, parçacığın o ana kadar elde ettiği en iyi uygunluk değeri olan *pbest* değeri ile karşılaştırılarak, *pbest* değeri güncellenir. Aynı şekilde, parçacığın o ana kadar sahip olduğu en iyi uygunluk değeri, sürünün o ana kadar elde ettiği en iyi uygunluk değeri olan *gbest* değeri ile karşılaştırılarak, *gbest* değeri güncellenir. Elde edilen yeni *pbest* ve *gbest* değerleri ve Eş. 2 yardımıyla yeni hız bilgisi hesaplanır. Eş. 3 kullanılarak yeni konum bilgisi elde edilir. Problem için verilen sonlandırma kriterine (maksimum iterasyon sayısı veya minimum hata koşulu) ulaşılıncaya kadar iterasyon devam eder.

2.3. Parçacık Sürü Optimizasyonuna Dayalı Kümeleme
(Particle Swarm Optimization Based Clustering)

İlk kez Omran vd. [27] tarafından tanıtılan PSO tabanlı kümeleme algoritmasında her bir parçacık, çözüm uzayında eş zamanlı olarak arama yapmakta ve her biri probleme çözüm olan aday çözümler tutmaktadır. Bu aday çözümler, konum bilgisi olan *c* adet küme merkez vektöründen oluşmaktadır. *d* boyutlu bir veri setinde *c.* kümenin merkezi $m_c = (m_{c1}, m_{c2}, \dots m_{cd})$ ve *i.* parçacığın konum bilgisi ise $x_i = (m_1, m_2, \dots m_c)$ şeklindedir. Her iterasyonda parçacığın hız bilgisi Eş. 2 yardımıyla güncellenmektedir. Elde edilen güncel hız vektörü konum vektörüne eklenerek, yeni konum

bilgisi bulunur ve parçacık çözüm uzayında o yöne doğru hareket eder. PSO tabanlı kümeleme algoritmasının sözde koduna Şekil 1'de yer verilmiştir.

Algoritma *p* tane parçacığa sahip olan bir sürüden oluşmaktadır. Başlangıçta bu *p* tane parçacığın her birine konum bilgisi olarak, rastgele *k* tane küme merkezi ve rastgele hız bilgisi atanmaktadır. Veri setinde yer alan tüm nesnelerin, her bir parçacığın konum bilgisi olarak tuttuğu küme merkezine olan Öklid uzaklığı hesaplanır. Nesneler, *k* tane küme merkezinden kendisine en yakın olan kümeye atanır. Bu işlemden sonra küme merkezleri tekrar hesaplanarak, yeni merkezler (konum bilgisi) uygunluk fonksiyonuna gönderilir. Elde edilen uygunluk değeri, parçacığın önceden sahip olduğu en iyi merkez bilgisi ile karşılaştırılarak güncelleme yapılır. Aynı şekilde *gbest* değerini güncellemek için, parçacığın o ana kadar sahip olduğu en iyi merkez bilgisi ile sürünün o ana kadar sahip olduğu en iyi merkez bilgisi karşılaştırılır ve gerekiyorsa güncelleme yapılır. Güncel *pbest* ve *gbest* değerleri kullanılarak, hız ve konum bilgileri sırasıyla tekrar hesaplanır. Maksimum iterasyon sayısına ulaşılıncaya kadar, bu işlemlere devam edilir ve sonucunda optimum kümeyi oluşturacak olan küme merkezleri elde edilir.

3. ÖNERİLEN ALGORİTMALAR (PROPOSED ALGORITHMS)

3.1. Başlangıç Merkezi Seçimi (Selecting Initial Cluster Centers)

Kümeler arasındaki farklılığın maksimum olması amaçlandığı için küme merkezlerinin birbirinden olabildiğince uzak seçilmesi hedeflenmiştir. Yaklaşımda, küme merkezi seçilirken, her parçacık için; *d* boyutlu *n* elemanlı bir veri setinde bulunan elemanlardan rastgele bir eleman seçilir. Seçilen bu eleman c_1 merkezi olarak atanır ve bu merkezin veri setindeki diğer tüm elemanlara olan Öklid uzaklığı bulunur. Bu elemanlar içerisinden merkeze en uzak olan eleman, c_2 merkezi olarak atanır. Sonra veri setindeki tüm elemanların (c_1 ve c_2 olarak seçilen merkez noktaları hariç) c_1 ve c_2 merkezlerine olan toplam uzaklıkları;

$$Sdist_{i3} = dist_{ic_1} + dist_{ic_2} \ \ (i = 1,2,\dots,n) \qquad (4)$$

Eş. 4 yardımıyla hesaplanır. Toplam uzaklığı en büyük olan eleman c_3 merkezi olarak atanır. Bu işlem, *k* kümeli bir veri setine uygulandığında c_k merkezine ulaşılıncaya kadar devam eder.

Bu yaklaşımımızda Başlangıç Merkezi Seçimi yöntemi, normalize edilmiş ve önerdiğimiz öznitelik seçimi yaklaşımına göre öznitelik indirgenmesi yapılmış veri setine uygulanmıştır. Önerilen öznitelik seçimi belirlenen eşik değerine göre farklı boyutlara indirgenerek yapılmıştır.

3.2. Öbek Tabanlı Yaklaşım (Coreset Based Approach)

Bu yaklaşımda, toplayıcı hiyerarşik kümeleme yöntemlerinden olan tek bağlantı algoritması ile birbirine yakın olan öğelerin aynı grupta toplanarak öbekler [29] halinde tek bir öğe gibi davranması sağlanmıştır. Bu yaklaşımdaki amaç, optimal çözüme ulaşılan iterasyon sayısını azaltmak ve birbirine benzeyen öğelerin birlikte hareket etmesine olanak sağlamaktır. Yaklaşım *n* elemanlı veri seti için uygulandığında, tüm elemanlar tek başına birer öbek olarak ele alınmış ve *n* öbek ile uygulamaya başlanmıştır. Her bir öbeğin merkezinin birbirlerine olan uzaklıkları hesaplanmış, *n*x*n* boyutlu bir uzaklık matrisi oluşturulmuştur. *n*x*n* boyutlu uzaklık matrisindeki her satırın ortalaması hesaplanarak *n*x*1* boyutlu ortalama matrisi

ALGORİTMA: PSO Tabanlı Kümeleme Algoritması

Girdi: ***p***, parçacık sayısı; ***k***, küme sayısı; ***N***, veri setindeki eleman sayısı; ***maxIterasyon***, maksimum iterasyon sayısı
Çıktı: Optimum küme merkezi

```
BEGIN
for i = 1 to p do
    rastgele hız bilgisi (v_i) ata
    k tane rastgele merkez bilgisi (x_i) ata
endfor
for iter = 1 to maxIterasyon do
    for i = 1 to p do
        for j = 1 to N do
            n_j nesnesinin x_i parçacığındaki küme merkezlerine uzaklığını hesapla
            n_j nesnesini kendisine en yakın merkeze sahip olan kümeye ata
        endfor
        f(x_i(t)) uygunluk değerini hesapla
        p_best ve g_best değerlerini güncelle
        i. parçacığın hız vektörünü hesapla
        i. Parçacığın konum vektörünü güncelle
    endfor
endfor
END
```

Şekil 1. PSO tabanlı kümeleme algoritması [28] (PSO-based clustering algorithm)

oluşturulur. Bu matrisin de ortalaması alınarak elde edilen değer, uzaklık dağılımlarına bakılarak elde edilen yüzde değerleri ile çarpılarak bir eşik değeri belirlenir. Uzaklık dağılımlarının algoritması Şekil 2'de verilmiştir. Bir sonraki adımda ise uzaklık matrisine bakılarak birbirine en yakın olan öğeler bulunarak birleştirilir, bir öbek oluşturulur. Bu öbeğin merkezi, o öbeğe dâhil olan öğelerin özniteliklerinin ayrı ayrı ortalamasının alınması ile hesaplanır. Öbek oluşturma işlemi sonrasında eleman sayısı azaldığı için uzaklık matrisi yeniden hesaplanır ve elde edilen bu matristen yine uzaklıkları en az olan çift bulunarak birleştirme işlemi uygulanır. Uzaklık matrisindeki minimum değer, eşik değerine ulaşıncaya kadar birleştirme işlemine devam edilir. Elde edilen öbeklerden birbirinden farklı olacak şekilde, küme sayısı kadar rastgele öbek seçilir ve kümeleme algoritmasına bu öbeklerin merkezi başlangıç küme merkezi olarak verilir. Kümeleme algoritmasının diğer işlem adımları da işletilerek algoritma sonlandırılır.

4. SONUÇLAR VE TARTIŞMALAR (RESULTS AND DISCUSSIONS)

Bu bölümde geliştirilen uygulamada kullanılan veri setlerinin tanıtılmasına, veri setlerine uygulanan ön işlemlere ve uygulamanın verimliliğinin test edilmesi için kullanılan Silhouette [30] ve Rand [31] küme değerlendirme indekslerine değinilmiştir. Algoritmaları durdurmak için kullanılan sonlandırma kriteri, K-ortalamalar algoritması için küme merkezlerinin değişmemesi; PSO tabanlı kümeleme algoritması için ise *gbest* değerinin yani küresel en iyinin değişmemesidir. Uygulamadaki kümeleme algoritmaları için kullanılan parametrelerin değerleri Tablo 1'deki gibidir. C# programlama dili kullanılarak geliştirilen bu uygulamadaki test sonuçları, sonuçlarının güvenilirliği için 100 kez çalıştırılarak elde edilmiştir. Sonuçlar, i7 1.90 GHz işlemcili, 6GB RAM'e ve Windows 8 işletim sistemine sahip bilgisayar üzerinde çalıştırılarak elde edilmiştir.

4.1. Veri Setleri (Datasets)

Geliştirilen yöntemler en sık kullanılan Breast Cancer Wisconsin, Contraceptive Method Choice (CMC), Heart Disease (Cleveland), Yeast veri seti ve bir adet Colon Cancer mikrodizi veri seti üzerinde test edilmiştir. Veri setlerinin genel özelliklerine Tablo 2'de yer verilmiştir.

4.2. Veri Setlerine Uygulanan Ön işlemler (Pre-processing on Datasets)

4.2.1. Normalizasyon (Normalization)

Uygulamada kullanılan veri setlerindeki nesnelerin değerleri arasındaki büyük farklılıkların azaltılması için, veri setlerine Eş. 5 kullanılarak normalizasyon işlemi uygulanmış ve veri setlerindeki nesnelerin değerleri [0,1] aralığına dönüştürülmüştür.

$$f'_{ij} = \frac{f_{ij} - f_{min}}{f_{max} - f_{min}} \quad (5)$$

Eşitlikte yer alan f_{ij}, veri setindeki *i.* nesnenin *j.* özelliğini (sample); f_{min} ve f_{max} değerleri, her özelliğin kendi içinde alınan minimum ve maksimum değerlerini göstermektedir.

4.2.2. Önerilen öznitelik seçimi yaklaşımı (Feature selection approach)

Öznitelik seçimi, veri kümesindeki öznitelikler arasından, yapılacak olan işlem için en uygun olan özniteliklerin seçilmesidir [32]. Yüksek boyutlu verilerin kümeleme

ALGORİTMA: Uzaklık Dağılımı Algoritması

Girdi: ***nxn*** uzaklık matrisi
Çıktı: ***th*** eşik değeri

1 BEGIN
2 Uzaklık matrisini uzaklık vektörüne çevir
3 Uzaklık vektörünü küçükten büyüğe doğru sırala
4 Vektördeki en küçük değeri minimum eşik değeri, en büyük değeri maksimum eşik değeri olarak belirle
5 Minimum ve maksimum eşik değeri aralığını 10 parçaya böl
6 5. Adımda elde edilen 10 eşik değeri için tek bağlantı yöntemini uygula
7 END

Şekil 2. Uzaklık dağılımının algoritması (Algorithm of distance distribution)

Tablo 1. Algoritmalarda kullanılan parametreler (Parameters used in algorithms)

PSO Tabanlı Kümeleme Algoritması	K-ortalamalar Algoritması
İterasyon sayısı: 100	İterasyon sayısı: 100
Uygunluk fonksiyonu: *Öklid uzaklığı*	Benzerlik parametresi: *Öklid uzaklığı*
Parçacık sayısı *(Particle):* 3	-
Bilişsel *(cognitive)* ve sosyal *(social)* parametreler *(c_1, c_2):* 2	-
Atalet değeri*(w):* 0.9	-
Algoritmayı sonlandırma kriteri: gbest değerinin değişmemesi	Algoritmayı sonlandırma kriteri: Küme merkezlerinin değişmemesi

Tablo 2. Kullanılan veri setleri ve veri setlerine öznitelik seçimi uygulanması sonucundaki değişimin karşılaştırılması (Used datasets and comparison of differences that occur after applying feature selection on datasets)

Veri seti	Veri sayısı (n)	Küme sayısı (k)	Öznitelik sayısı (d)	Öznitelik seçiminden sonraki öznitelik sayısı (d')
Breast Cancer	699	2	9	5
CMC	1473	3	9	5
Heart Disease	303	5	13	8
Yeast	1484	10	8	4
Colon Cancer	2000	2	62	50

algoritmalarında kullanımı hem algoritmanın yavaş çalışmasına hem de verilerin kümelenmesini kolaylaştıracak olan özniteliklerin, gereksiz öznitelikler yüzünden anlaşılmamasına neden olmaktadır [33]. Bu uygulama kapsamında kullanılan büyük boyutlu normalize edilmiş veri setleri üzerinde, doğru kümeleme yapılabilmesi amacıyla öznitelik seçimi işlemi için bir yaklaşım önerilmiştir. Önerilen öznitelik seçimi yaklaşımı için, veri setindeki her özelliğin kendi içerisinde standart sapması (σ_{fj}) ve varyansı ($var(f_j)$) hesaplanarak Eş. 6'da yerlerine yazılıp bir indirgenme değeri (RV-Reduction Value) oluşturulmuştur. Eş. 6'da d, öznitelik sayısını; μ_{fj} ve $var(f_j)$ sırasıyla j. özelliğin ortalamasını ve varyansını göstermektedir. Önerilen öznitelik seçiminin kullanılacak olan veri setlerine uygulanması sonucunda elde edilen değerlere Tablo 2'de yer verilmiştir.

$$RV = \left[\frac{1}{d}\sum_{j=1}^{d}\left(100.\frac{var(f_j)}{\mu_{f_j}}\right)\right] - 1 \quad (6)$$

İndirgenme değerinin bulunmasını sağlayan eşitlik, deneysel çalışmalar sonucunda elde edilmiştir.

4.3. Küme Değerlendirme İndeksleri (Cluster Validation Indexes)

Küme değerlendirme, kümeleme sonuçlarının karşılaştırılması ve sonuçların doğrulanması açısından kümeleme analizi için önemlidir. Kullanmış olduğumuz 4 veri setinin küme etiketleri önceden bilindiği için küme dışı değerlendirme indeksi olan Rand indeks, mikrodizi veri seti için ise Silhouette indeksi doğruluk indeksi olarak kullanılmıştır. Y, kümeleme algoritmasının işletilmesi sonucunda elde edilen küme etiketleri; Z, dışarıdan verilen küme etiketleri olmak üzere; x_i ve x_j, X kümesine ait iki nesne olsun. Rand indeks, Y ve Z küme etiketleri için aynı grupta ya da farklı grupta yer alan nesnelerin önemini belirtmektedir. Eş. 7 kullanılarak hesaplanan bu indeks, [0,1] aralığında değer almaktadır. Eşitlikteki a değeri; x_i ve x_j nesnelerinin Y küme etiketlerinde aynı kümede ve Z küme etiketlerinde aynı kümede olma durumunun sayısıdır. b değeri; x_i ve x_j nesnelerinin Y küme etiketlerinde aynı kümede, ancak Z küme etiketlerinde farklı kümede olma durumunun sayısıdır. c değeri; x_i ve x_j nesnelerinin Yküme etiketlerinde farklı kümede, ancak Zküme etiketlerinde aynı kümede olma durumunun sayısıdır. d değeri; x_i ve x_j nesnelerinin Yküme etiketlerinde farklı kümede ve Zküme etiketlerinde farklı kümede olma durumunun sayısıdır.

$$RI = \frac{a+d}{\mathrm{a+b+c+d}} \quad (7)$$

Silhouette indeksi, her bir noktanın diğer kümelerdeki noktalara oranla kendi kümesindeki noktalara olan benzerliğinin ölçülmesine yardımcı olmaktadır. Bu indeks Eş. 8 yardımıyla hesaplanmıştır. Burada $S(i)$, X_j kümesine ait olan i. noktanın Silhouette indeksini; $a(i)$, X_j kümesindeki tüm elemanların i noktasına olan Öklid uzaklıklarının ortalamasını; $b(i)$, i noktasının Xj kümesi dışında tüm kümelerdeki (X_m (m=1, … k ve $m\neq j$)) elemanlara olan küme bazındaki ortalama uzaklıkların minimumunu ifade etmektedir. Bu indeks [−1,1] aralığında değer almaktadır. Bu değerin 1'e yakın olması kümelemenin iyi, −1'e yakın olması kümelemenin kötü olduğunun bir göstergesidir.

$$S(i) = \frac{b(i)-a(i)}{\max\{a(i),b(i)\}} \quad (8)$$

Diğer performans kriteri ise çözüme ulaşılan ortalama iterasyon sayısıdır. Çözüme ulaşılan ortalama iterasyon sayısının küçük olması kullanılan yöntemin iyi kümeleme yaptığını göstermektedir.

4.4. Bulgular (Results)

Bu kısımda, normalize edilmiş veri setleri ve öznitelik seçimi yapılmış normalize (FSN-Feature Selection Normalized) veri setleri üzerinde çalıştırılan PSO tabanlı kümeleme ve K-ortalamalar kümeleme algoritmalarının sonuçlarına yer verilmiştir. *PSOBMS* olarak isimlendirilen yöntem, "*Başlangıç merkezi seçimi (BMS)"* isimli yaklaşımdan elde edilen küme merkez bilgilerinin PSO tabanlı kümeleme algoritmasına başlangıç küme merkezi olarak gönderilerek, PSO tabanlı kümeleme algoritmasının ilgili veri seti üzerinde çalıştırılmasını gösterir. *"Öbek tabanlı yaklaşım (ÖTY)"*'ın PSO tabanlı kümeleme algoritmasına uyarlandığı yöntem *PSOÖTY* olarak isimlendirilmiştir. Aynı şekilde, K-ortalamalar algoritmasının başlangıç küme merkezi seçimi *BMS* isimli yaklaşım ile gerçekleniyorsa, bu yöntem *K-*

meansBMS olarak isimlendirilir. Tablo 3'te Breast Cancer veri seti üzerinde, geliştirilen yaklaşımların ve standart kümeleme yöntemlerinin uygulanması sonucunda elde edilen değerler yer almaktadır. Önerilen *PSOÖTY* algoritmasının, standart PSO tabanlı kümeleme algoritmasına kıyasla FSN Breast Cancer veri seti için Rand indeksinin ve ortalama iterasyon sayısının daha iyi olduğu gözlemlenmiştir. Aynı veri seti için K-ortalamalar algoritmalarına bakıldığında, önerilen *K-MeansBMS* algoritmasının standart K-ortalamalar algoritmasından Rand indeks bakımından daha iyi sonuç verdiği görülmüştür. Normalize Breast Cancer veri seti için ise PSO tabanlı kümeleme yaklaşımlarında, önerilen *PSOÖTY* algoritmasının ortalama iterasyon sayısının standart PSO algoritmasına kıyasla daha iyi olduğu gözlemlenmiştir. Aynı veri seti için K-ortalamalar algoritmalarına bakıldığında; önerilen *K-meansBMS* algoritmasının Rand indeksinin, standart K-ortalamalar algoritmasına kıyasla daha iyi olduğu görülmektedir.

FSN Breast Cancer veri seti üzerinde işletilen PSO algoritmalarının, Normalize Breast Cancer veri seti üzerinde işletilen PSO algoritmalarına kıyasla Rand indeksi açısından daha düşük sonuçlar verdiği gözlenmiştir (*PSOÖTY* algoritması hariç). Bunun sebebi ise, Breast Cancer veri setinde yer alan 16 adet kayıp verinin bulunduğu, anlamlı olan öznitelik sütununun indirgenmesidir. Tablo 3 için K-ortalamalar ve PSO algoritmaları, FSN veri seti ve Normalize veri seti olarak karşılaştırıldığında; en iyi iterasyon sayısının ve Rand indeksin, önerilen *PSOÖTY* algoritmasının işletilmesi sonucunda elde edilen değerler olduğu gözlemlenmiştir. CMC veri seti üzerinde uygulanan yaklaşımların sonuçlarına Tablo 4'te yer verilmiştir. Geliştirilen *PSOBMS* ve *K-meansBMS* algoritmasının Normalize CMC veri seti üzerinde elde ettiği Rand indeksi standart yöntemlere kıyasla daha iyi çıkmıştır. *PSOÖTY* algoritmasının Normalize CMC ve FSN CMC veri seti üzerinde işletilmesi, standart PSO ve K-ortalamalar kümeleme algoritmalarına göre daha az iterasyonda çözüme ulaşılmasını sağlamıştır. K-ortalamalar ve PSO algoritmaları, FSN veri seti ve Normalize veri seti olarak karşılaştırıldığında; en iyi Rand indeksin, önerilen *PSOBMS* algoritmasının Normalize CMC veri seti üzerinde işletilmesi sonucunda elde edildiği ve en iyi iterasyon sayısının ise yine önerilen *PSOÖTY* algoritmasının FSN CMC veri seti üzerinde işletilmesi sonucunda elde edildiği gözlemlenmiştir.

Heart veri seti üzerinde önerilen yaklaşımların uygulanması sonucu elde edilen değerler Tablo 5'te yer almaktadır. Normalize veri seti üzerinde işletilen PSO tabanlı kümeleme yaklaşımlarından elde edilen bulgularda, önerilen *PSOBMS* algoritmasının standart PSO tabanlı kümeleme algoritmasına kıyasla daha iyi olduğu gözlemlenmiştir. K-ortalamalar

Tablo 3. Breast Cancer veri seti üzerine uygulanan önerilen algoritmaların standart versiyonları ile karşılaştırılması (Comparison of proposed algorithms with their standard versions on Breast Cancer dataset)

BREAST	Performans Kriteri	PSO	PSOBMS	PSOÖTY (TH=1,0943)	K-means	K-meansBMS
Normalize Veri	*Rand İndeks*	0,9116	0,9106	0,8272	0,9052	0,9125
	Ort. İter. Say.	4,2	5,29	2	8,38	8,49
Öznitelik Seçimi Yapılmış Normalize Veri	*Rand İndeks*	0,8959	0,8939	0,9177	0,8863	0,8920
	Ort. İter. Say.	3,76	3,95	2	6,15	4,71

Tablo 4. CMC veri seti üzerine uygulanan önerilen algoritmaların standart versiyonları ile karşılaştırılması (Comparison of proposed algorithms with their standard versions on CMC dataset)

CMC	Performans Kriteri	PSO	PSOBMS	PSOÖTY (TH=0,0)	K-means	K-meansBMS
Normalize Veri	*Rand İndeks*	0,5490	0,5531	0,5452	0,5442	0,5515
	Ort. İter. Say.	8,28	12,67	7,78	11,27	12,37
Öznitelik Seçimi Yapılmış Normalize Veri	*Rand İndeks*	0,5477	0,5362	0,5141	0,5345	0,5316
	Ort. İter. Say.	5,03	4,47	3,97	15,45	5,65

Tablo 5. Heart veri seti üzerine uygulanan önerilen algoritmaların standart versiyonları ile karşılaştırılması (Comparison of proposed algorithms with their standard versions on Heart dataset)

HEART	Performans Kriteri	PSO	PSOBMS	PSOÖTY (TH=0,1536)	K-means	K-meansBMS
Normalize Veri	*Rand İndeks*	0,6347	0,6437	0,6358	0,6390	0,6494
	Ort. İter. Say.	7,62	7,72	7,98	8	7,7
Öznitelik Seçimi Yapılmış Normalize Veri	*Rand İndeks*	0,6328	0,6423	0,6368	0,6330	0,6456
	Ort. İter. Say.	7,55	7,29	7,43	9,52	7,33

Tablo 6. Yeast veri seti üzerine uygulanan önerilen algoritmaların standart versiyonları ile karşılaştırılması (Comparison of proposed algorithms with their standard versions on Yeast dataset)

YEAST	Performans Kriteri	PSO	PSOBMS	PSOÖTY (TH=0,0096)	K-means	K-meansBMS
Normalize Veri	*Rand İndeks*	0,7499	0,6915	0,7502	0,7484	0,6207
	Ort. İter. Say.	13,78	18,21	13,53	29,42	100
Öznitelik Seçimi Yapılmış Normalize Veri	*Rand İndeks*	0,7413	0,7382	0,7413	0,7399	0,7283
	Ort. İter. Say.	16,15	20	13,74	30,65	100

Tablo 7. Colon Cancer veri seti üzerine uygulanan önerilen algoritmaların standart versiyonları ile karşılaştırılması (Comparison of proposed algorithms with their standard versions on Colon Cancer dataset)

COLON CANCER	Performans Kriteri	PSO	PSOBMS	PSOÖTY (TH=1,8379)	K-means	K-meansBMS
Normalize Veri	*Silhouette İndeks*	0,3754	0,4850	0,6481	0,4891	0,4883
	Ort. İter. Say.	3,35	7,88	2,07	11,11	10,09
Öznitelik Seçimi Yapılmış Normalize Veri	*Silhouette İndeks*	0,3800	0,4851	0,6643	0,4948	0,4975
	Ort. İter. Say.	3,32	8,15	2	11	8,01

algoritmalarının aynı veri seti üzerinde çalıştırılması sonucunda elde edilen bulgular ise önerilen *K-meansBMS* algoritmasının standart K-ortalamalar algoritmasına kıyasla daha iyi olduğunu göstermektedir. FSN Heart veri seti incelendiğinde; önerilen *PSOBMS* algoritmasının gerek Rand indeksi gerekse iterasyon sayısı bakımından standart PSO tabanlı kümeleme algoritmasına oranla daha iyi sonuçlar verdiği görülmektedir. Aynı şekilde, aynı veri seti üzerinde işletilen K-ortalamalar algoritmalarından elde edilen en iyi değerler, önerilen *K-meansBMS* algoritması tarafından bulunmuştur. Her iki veri setini de değerlendirecek olursak; *K-meansBMS* algoritmasının Rand indeksinin daha yüksek olduğu yani daha doğru kümeler elde ettiği Tablo 5'te açıkça görülmektedir. Tablo 6'da verilen değerler FSN Yeast ve Normalize Yeast veri seti üzerinde işletilmiş olan yaklaşımlara aittir. Tabloya bakıldığında önerilen *PSOÖTY* ile standart PSO tabanlı kümeleme algoritmasının FSN Yeast veri seti üzerinde işletilmesi sonucunda elde edilen Rand indekslerin birbiri ile aynı olduğu gözlemlenmiştir. Ancak geliştirilen *PSOÖTY* algoritmasının daha az iterasyon ile bu sonuca ulaşması, geliştirilen algoritmanın standart PSO tabanlı kümeleme algoritmasından daha iyi olduğunun bir göstergesidir.

Normalize veri setine bakıldığında; PSO tabanlı kümeleme algoritmaları arasında, hem Rand indeks hem de iterasyon sayısı bakımından en iyi sonucu veren algoritmanın önerilen *PSOÖTY* algoritması olduğu gözlemlenmektedir. FSN Yeast veri seti ile Normalize Yeast veri seti kıyaslandığında; Rand indeks değerinin ve ortalama iterasyon sayısının Normalize Yeast veri seti üzerinde işletilen *PSOÖTY* kümeleme algoritmasında iyi sonuçlar verdiği gözlemlenmiştir. Tablo 7, Colon Cancer veri setine uygulanan yaklaşımların test sonuçlarını göstermektedir. Colon Cancer veri setinin küme etiketleri önceden bilinmediği için doğruluk indeksi olarak Silhouette indeksi baz alınmıştır. FSN Colon Cancer veri seti üzerinde, standart PSO tabanlı kümeleme algoritmasının ve geliştirilen PSO tabanlı kümeleme yaklaşımlarının işletilmesi sonucunda elde edilen en iyi Silhouette indeksi ve çözüme ulaşılan ortalama iterasyon sayısı *PSOÖTY*'na aittir. Yine aynı veri seti üzerinde işletilen standart K-ortalamalar ve *K-meansBMS* algoritması kıyaslandığında; en iyi Silhouette indeks ve ortalama iterasyon sayısının önerilen *K-MeansBMS* algoritmasının işletilmesi sonucunda elde edildiği gözlemlenmiştir. Normalize Colon Cancer veri setine bakıldığında, geliştirilen yaklaşımlardan *PSOÖTY* algoritmasının Silhouette indeksinin ve ortalama iterasyon sayısının standart K-ortalamalar ve standart PSO tabanlı kümeleme algoritmalarına kıyasla oldukça yüksek bir başarı elde ettiği görülmektedir. FSN ve Normalize Colon Cancer veri seti karşılaştırıldığında ise önerilen *PSOÖTY* algoritmasının gerek daha doğru kümeleme yapma, gerekse çözüme daha az iterasyonda ulaşma konusunda standart yöntemlere kıyasla FSN Colon Cancer veri seti üzerinde daha başarılı olduğu görülmektedir. Ayrıca, *PSOÖTY* algoritmasının işletilmesi sonucunda bazı veri setlerinde ortalama iterasyon sayısının "2" çıkmasının nedeni, başlangıçta elde edilen öbek sayısının küme sayısına eşit olmasıdır.

5. SONUÇLAR (CONCLUSIONS)

Bu çalışmada gerçekleştirilen yaklaşımlar ile çok boyutlu veriler daha yüksek doğrulukla kümelenmiştir. Bu amaç doğrultusunda, yaygın olarak kullanılan kümeleme yöntemlerinden K-ortalamalar ve PSO tabanlı kümeleme algoritmaları temel alınmıştır. Literatürde en sık kullanılan 4 adet veri seti ve bir adet mikrodizi veri seti için K-ortalamalar ve PSO tabanlı kümeleme algoritmalarının başlangıç küme merkezi seçimine yeni yaklaşımlar getirilerek kümeleme yapılmış ve sonuçlarının doğruluğunun test edilmesi için Rand ve Silhouette küme

doğruluk indeksleri kullanılmıştır. Ayrıca öbek (coreset) yaklaşımı PSO tabanlı kümeleme algoritmasına uyarlanarak aynı testler gerçekleştirilmiştir. Normalize Breast Cancer veri seti için yapılan deneysel çalışmalar sonucunda elde edilen bulgular; en iyi Rand indeks doğruluk değerinin önerilen *K-meansBMS* algoritması tarafından elde edildiğini; en iyi ortalama iterasyon sayısının ise önerilen *PSOÖTY* algoritmasının işletilmesi sonucunda elde edildiğini göstermektedir. Normalize CMC veri seti için elde edilen bulgularda ise; en iyi Rand indeks doğruluk değerinin önerilen *PSOBMS* algoritması tarafından elde edildiği ve en iyi sonuca ulaşılan en iyi ortalama iterasyon sayısının yine önerilen *PSOÖTY* algoritması tarafından elde edildiği gözlemlenmiştir. Normalize Heart veri seti için alınan sonuçlarda en iyi Rand indeks değerine, önerilen *K-meansBMS* algoritması tarafından ulaşılmıştır. Aynı veri seti için en iyi ortalama iterasyon sayısını veren algoritma, standart PSO algoritmasıdır. Normalize Yeast veri seti için en iyi Rand indeks değerinin ve ortalama iterasyon sayısının elde edildiği algoritma önerilen *PSOÖTY* algoritmasıdır. Mikrodizi veri seti olan ve küme etiketleri önceden bilinmeyen Colon Cancer veri setinin normalize edilmiş hali için alınan test sonuçlarında; gerek Silhouette indeksi bakımından, gerekse ortalama iterasyon sayısı bakımından en iyi sonuca önerilen *PSOÖTY* algoritması tarafından ulaşılmıştır.

FSN Breast Cancer veri seti için yapılan deneysel çalışmalar sonucunda elde edilen bulgular; en iyi Rand indeks doğruluk değerinin ve en iyi ortalama iterasyon sayısının önerilen *PSOÖTY* algoritması tarafından elde edildiğini göstermektedir. FSN CMC veri seti için elde edilen bulgular; en iyi Rand indeks doğruluk değerinin standart PSO algoritması tarafından elde edildiğini ve en iyi ortalama iterasyon sayısının yine önerilen *PSOÖTY* algoritması tarafından elde edildiğini göstermektedir. FSN Heart veri seti için alınan sonuçlarda, en iyi Rand indeks değerine, önerilen *K-meansBMS* algoritması ile ulaşılmaktadır. Aynı veri seti için en iyi ortalama iterasyon sayısına ise önerilen *PSOBMS* algoritması ile ulaşılmaktadır. FSN Yeast veri seti için en iyi Rand indeks değerinin ve en iyi ortalama iterasyon sayısının elde edildiği algoritma önerilen *PSOÖTY* algoritmasıdır. FSN Colon Cancer veri seti için alınan sonuçlarda; hem en iyi Silhouette indeksi hem de en iyi ortalama iterasyon sayısı bakımından en iyi sonucu veren algoritmanın önerilen *PSOÖTY* algoritması olduğu gözlemlenmiştir.

Normalize veri setleri ve öznitelik seçimi yapılmış normalize veri setleri üzerinde test edilen algoritmaların sonuçlarına genel olarak bakıldığında; önerilen yaklaşımların, standart K-ortalamalar ve standart PSO tabanlı kümeleme algoritmalarının sonuçlarına kıyasla daha iyi kümeleme yaptığı görülmektedir. Diğer bir deyişle, önerilen *PSOÖTY* yaklaşımının çoğu veri setinde hem Rand/Silhouette doğruluk indeksi ve hem de ortalama iterasyon sayısı bakımından en iyi sonuçları verdiği görülmektedir. Önerilen yaklaşımlardan biri olan *PSOÖTY* algoritması ile Colon Cancer mikrodizi veri seti için oldukça yüksek bir başarım elde edilmiştir. Gerçekleştirilen deneylerle bu performans artışı gözler önüne serilmiştir.

KAYNAKLAR (REFERENCES)

1. Bertone P., Gerstein M., Integrative Data Mining: The New Direction in Bioinformatics Machine Learning for Analyzing Genome-wide Expression Profiles, IEEE Engineering in Medicine and Biology, 20, 33-40, 2001.
2. Valafar F., Pattern Recognition Techniques in Microarray Data Analysis: A Survey, Annals of New York Academy of Sciences, 980 (1), 41-64, 2002.
3. Jiang D., Tang C., Zhang A., Cluster Analysis for Gene Expression Data: A Survey, IEEE Transactions on Knowledge and Data Engineering, 16 (11), 1370-1386, 2004.
4. Handl J., Knowles J., Kell D.B., Computational Cluster Validation in Post-Genomic Data Analysis, Bioinformatics, 21, 3201–3212, 2005.
5. Esmin A.A.A., Coelho R.A., Matwin S., A review on particle swarm optimization algorithm and its variants to clustering high-dimensional data, Artificial Intelligence Review, 44 (1), 23-45, 2015.
6. Krishnasamy G., Kulkarni A.J., Paramesran R., A hybrid approach for data clustering based on modified cohort intelligence and K-means, Expert Systems with Applications, 41 (13), 6009-6016, 2014.
7. Ekinci S., Hekimoğlu B., Multi-machine power system stabilizer design via HPA algorithm, Journal of the Faculty of Engineering and Architecture of Gazi University, 32 (4), 1271-1285, 2017.
8. Alataş B., Özer A.B., Mining of generalized interesting classification rules with artificial chemical reaction optimization algorithm, Journal of the Faculty of Engineering and Architecture of Gazi University, 32 (1), 101-118, 2017.
9. Kar A.K., Bio Inspired Computing–A Review of Algorithms and Scope of Applications, Expert Systems with Applications, 59, 20-32, 2016.
10. Forgy E.W., Cluster Analysis of Multivariate Data: Efficiency vs. Interpretability of Classifications, Biometrics, 21 (3), 768-769, 1965.
11. Kaufman L. ve Rousseeuw, P.J., Finding Groups in Data: An Introduction to Cluster Analysis, John Wiley & Sons, New York, 1990.
12. Katsavounidis I., Kuo C., Zhang Z., A New Initialization Technique for Generalized Lloyd Iteration, IEEE Signal Processing Letters, 1 (10), 144 -146, 1994.
13. Bradley P.S., Fayyad U.M., Refining Initial Points for K -Means Clustering, 15th International Conference on Machine Learning, San Francisco-ABD, 91-99, 1998.
14. Khan S.S., Ahmad A., Cluster Center Initialization Algorithm for K-means Clustering, Pattern recognition letters, 25 (11), 1293-1302, 2004.
15. Arai K., Barakbah A.R., Hierarchical K-means: An Algorithm for Centroids Initialization for K-means, Reports of the Faculty of Science and Engineering Saga University, 36 (1), 25-31, 2007.

16. Erişoğlu M., Çalış N., Sakallıoğlu S., A New Algorithm for Initial Cluster Centers in K-means Clustering, Pattern Recognition Letters, 32 (14), 1701-1705, 2011.

17. Aggarwal N., Aggarwal K., A Mid-Point Based K-means Clustering Algorithm for Data Mining, International Journal on Computer Science and Engineering (IJCSE), 4 (6), 1174-1180, 2012a.

18. Aldahdooh R.T., Ashour W., DIMK-means 'Distance-based Initialization Method for K-means Clustering Algorithm', International Journal of Intelligent Systems and Applications, 5 (2), 41-51, 2013.

19. Qiao J., Lu Y., A new algorithm for choosing initial cluster centers for k-means, 2nd International Conference on Computer Science and Electronics Engineering (ICCSEE 2013), Paris-Fransa, 527-530, 2013.

20. Jothi R., Mohanty S.K., Ojha A., On Careful Selection of Initial Centers for K-means Algorithm, In Proceedings of 3rd International Conference on Advanced Computing, Networking and Informatics (ICACNI), 435-445, 2016.

21. Cui X., Potok T.E., Document clustering analysis based on hybrid PSO + K-means algorithm, Journal of Computer Sciences, 27-33, 2005.

22. Cui X., Potok T.E., Palathingal P., Document clustering using particle swarm optimization, In Swarm Intelligence Symposium, Kaliforniya-ABD, 185-191, 8-10 Haziran, 2005.

23. Dey L., Mukhopadhyay A., Microarray gene expression data clustering using PSO based K-means algorithm, UACEE International Journal of Computer Science and its Applications, 1 (1), 232-236, 2009.

24. Kennedy J., Eberhart R., Particle Swarm Optimization, In Proceedings of IEEE International Conference on Neural Networks, 4, 1942-1948, 1995.

25. Haltaş A., Alkan A., Karabulut M., Performance analysis of heuristic search algorithms in text classification, Journal of the Faculty of Engineering and Architecture of Gazi University, 30 (3), 417-427, 2015.

26. Poli R., Kennedy J., Blackwell T., Particle Swarm Optimization an Overview, Swarm Intelligence, 1 (1), 33–57, 2007.

27. Omran M., Salman A., Engelbrecht A.P., Image Classification Using Particle Swarm Optimization, In Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution and Learning (SEAL 2002), Singapur, 370-374, 2002.

28. Abraham A., Das S., Roy S., Swarm Intelligence Algorithms for Data Clustering, Soft Computing for Knowledge Discovery and Data Mining, Springer, Boston, ABD, 1, 279-313, 2008.

29. Bādoiu M., Har-Peled S., Indyk P., Approximate Clustering via Core-sets, In Proceedings of the 34th Annual ACM Symposium on Theory of Computing (STOC), Quebec-Canada, 250-257, 19-21 Mayıs 2002.

30. Rousseeuw P.J., Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis, Computational and Applied Mathematics, 20 (1), 53–65, 1987.

31. Rand W.M., Objective Criteria for the Evaluation of Clustering Methods, Journal of the American Statistical Association, 66 (336), 846–850, 1971.

32. Var E., Inan A., Differentially private attribute selection for classification, Journal of the Faculty of Engineering and Architecture of Gazi University 33 (1), 323-336, 2018.

33. Guyon I., Gunn S., Ben-Hur A., Dror G., Result Analysis of the NIPS 2003 Feature Selection Challenge, In Advances in Neural Information Processing Systems 17, 545–552, 2005.