**YZM 4008**

**Veri Madenciliği Dersi Proje Raporu**

**A picture containing symbol, graphics, circle, logo

Description automatically generated**

**Algoritma Adı: LRFDB-COA (Lévy Roulette FDB-COA)**

**Öğrenci Adı ve Soyadı: Veli Can Kurt**

**Öğrenci No: 397237**

**Öğrenci İletişim Bilgileri**

**Cep tel: 0530 118 24 07**

**\*e-mail adresi: velicankurt6@gmail.com**

**Özet:** Boyut azaltma, sınıflandırma problemlerinde kullanılan önemli yöntemlerden biridir. Yüksek boyutlu yani çok sayıda bağımsız değişken barındıran veri setlerinde, nitelik sayısının fazla olması algoritma performansını olumsuz etkileyebilir. Bu sebeple boyut azaltma teknikleri kullanılır. Kullanılan bu yöntem, veri setindeki nitelik sayısını azaltmayı ve algoritma performansını yükseltmeyi ve algoritmanın hesaplanma karmaşıklığını azaltmayı hedefler. Boyut azaltma, neredeyse tüm algoritmalar için performansı olumlu etkileyen bir faktördür. Bu nedenle, sınıflandırma problemlerinde boyut azaltma çalışmaları gerçekleştirilmektedir. Karmaşık arama uzaylarında en iyi çözümü bulmak neredeyse imkansızdır. Bu nedenle, bu tür sınıflandırma problemleri için meta-sezgisel arama algoritmalarından yararlanılmaktadır. Bu çalışmada, boyut azaltma için COA (Coyote Optimization Algorithm) algoritmasının iyileştirilmiş versiyonu olan LRFDB-COA algoritması, meta-sezgisel nitelik seçimi yöntemi olarak uyarlanmıştır. Bu yöntem, niteliklerin ağırlıklandırılmasını ve ağırlıkları eşik değerin altında kalan niteliklerin tespit edilmesini ve böylece bu niteliklerin veri setinden çıkarılarak veri setinin sınıflandırma problemi için daha sade olmasını sağlamaktadır. Bu algoritma, UCI Machine Learning veri havuzunda yer alan Urban Land Cover veri seti üzerinde test edilmiştir. Yapılan deneysel çalışmalar, LRFDB-COA tabanlı nitelik seçim yönteminin sınıflandırma problemleri için etkili bir araç olduğunu göstermektedir. Elde edilen sonuçlar, LRFDB-COA algoritmasının veri analizi ve makine öğrenmesinde boyut azaltma konusunda etkili bir yol sunduğunu göstermektedir.

**Anahtar Kelimeler:**  k-nn, COA, FDB, Lévy Flight, Boyut Azaltma, Meta-Sezgisel Ağırlıklandırma

**1. Giriş**

Yüksek boyutlu veri setlerinden anlamlı bilgilerin çıkarılmasında kritik bir rol oynayan boyut azaltma, veri madenciliği alanında büyük öneme sahiptir. Hesaplama, donanım ve doğru modelleme gibi zorlukları beraberinde getiren, giderek artan büyüklükteki ve karmaşıklıktaki veri setleriyle çalışırken, temel niteliklerin hedef parametre üzerindeki etkilerini belirlemek önemlidir. Bu yöntem sayesinde, hedef parametre üzerinde yeterince etkisi olmayan nitelikler veri setinden çıkarılarak daha anlaşılır bir veri seti elde edilebilir. Boyut azaltma yöntemleri, bu zorlukları aşmanın yanı sıra veri içerisindeki önemli yapıları ve ilişkileri korur, böylece modelleme sürecinde yapılan hataların tespit edilmesine ve düzeltilmesine yardımcı olur. En az sayıda nitelik içeren optimize edilmiş bir model, kabul edilebilir bir sınıflandırma performansı sunar. Bu nedenle, boyut azaltma çalışmaları, problem boyutu ve karmaşıklık seviyesi göz önünde bulundurularak meta-sezgisel arama algoritmalarının kullanılmasını gerektirir.

Meta-sezgisel arama algoritmaları, çözüm uzayında olasılık temelli ancak bilinçli bir şekilde arama gerçekleştiren yöntemleri içerisinde barındırmaktadır. Bu yöntemler her adımda oluşturulan çözüm kümesinden yola çıkarak yeni çözümler üretmektedir. Böylece arama uzayının en uygun çözümüne yani global optimuma yakın olan noktalarında aramalar yapılarak, yerel çözüm tuzaklarından da kurtularak en uygun çözüme ulaşmaya çalışılır [1]. Meta-Sezgisel arama algoritmalarının performansları probleme bağımlıdır. Bu doğada da bu şekildedir. Doğada her canlı ya da süreç kendi işleyişinde kusursuz bir performans sergilerken bir başka görev ya da işleyiş için çok başarısız olabilmektedir. Bunun yanında son dönemde geliştirilen modern MSA algoritmaları pek çok tipte farklı özellikteki optimizasyon problemleri üzerinde test edilerek geliştirildikleri için nispeten daha kararlı ve başarılı bir arama performansına sahip olmaktadırlar.

Meta-Sezgisel arama algoritmalarının yanında k-nn sınıflandırıcısının da boyut indirgeme yönteminde kullanılması, boyut indirgeme yöntemini başarılı hale getirmektedir. Bu projede COA (Coyote Optimization Algorithm) algoritmasının Lévy Flight ve FBD tabanlı iyileştirilmiş hali olan LRFDB-COA (Lévy Roulette FDB-COA) algoritması kullanılmıştır. Bu algoritma, k-nn sınıflandırıcıyla birleştirilerek melez bir algoritma elde edilmiştir. Bu melez algoritma, probleme ait niteliklerin ağırlıklandırılması ve sezgisel sınıflandırma gibi farklı gereksinimlere cevap veren yeteneklere sahip olarak geliştirilmiştir.

**2. Materyal ve Yöntem**

Geliştirilen melez algoritmanın anlaşılmasına yardımcı olması için üç alt bölüm hazırlanmıştır. Takip eden bölümde k-nn algoritması hakkında bilgi verilmektedir.

**2.1. k-nn (En Yakın Komşu) Algoritması**

k-nn, örnek tabanlı bir sınıflandırma algoritmasıdır. Algoritmanın temeli, sınıfı belirlenecek olan bir gözlem için problem uzayını temsil eden örnek gözlemler arasından benzer özelliklere sahip olanların referans alınmasına dayanır. *K*-nn sınıflandırıcı bir öğrenme yöntemi içermediği gibi probleme ait bir model de geliştirmez. *K-nn* algoritması, bir gözlemi sınıflandırmak için problem uzayını temsil eden örnek gözlemler arasından benzer özelliklere sahip olanları kullanır.

Aşağıda *K-nn* algoritmasının adımları verilmiştir.

1. *k* değerinin belirlenmesi: *K-nn* algoritmasının en önemli parametrelerinden biri *k* değeridir. *k*, komşu sayısını temsil eder ve sınıflandırma yaparken kaç komşunun dikkate alınacağını belirler. *k* değeri, problem ve veri setine bağlı olarak seçilir. Önceden belirlenmiş bir değer kullanılabileceği gibi, çapraz doğrulama veya optimize edici tekniklerle de belirlenebilir.
2. Uzaklık ölçümü: *K-nn* 'de, gözlemler arasındaki benzerlik veya uzaklık hesaplanır. Genellikle Öklid, Manhattan veya Minkowski gibi uzaklık metrikleri kullanılır. Bu metrikler, gözlemler arasındaki farkları ölçer ve benzerlik ölçüsü olarak kullanılır.
3. En yakın komşuların seçilmesi: Her gözlem için, hesaplanan uzaklık değerlerine göre en yakın *k* komşu seçilir.
4. Sınıflandırma: Seçilen en yakın komşuların sınıfları dikkate alınarak gözlem sınıflandırılır. Çoğunluk oylaması yöntemi kullanılabilir, yani en sık sınıf etiketi oylama yoluyla seçilebilir. Ayrıca, komşulara ağırlıklandırma faktörleri de uygulanabilir, yani komşuların uzaklıklarına göre ağırlıklar atanabilir.

Aşağıda *k* değerini ve veri setini girdi olarak alan k-nn algoritmasının sözde kodu verilmiştir.

**Algoritma 1.** *k*-nn algoritmasının sözde kodu [2]

|  |  |
| --- | --- |
| **Başla** | |
| 1. | Sınıf etiketi belirlenecek olan *q* sorgu gözlemini tanımla |
| 2. | **for i=1: n**     D[i]=*q* ile **Xi** arasındaki uzaklığı hesapla  **end** |
| 3. | Xq[k]=D[i] uzaklık dizisinden *q* sorgu gözlemine en yakın *k*-adet gözlemi belirle |
| 4. | Xq[k] gözlemlerinin sınıflarını dikkate alarak çoğunluk oylaması/ağırlıklı  oylama yöntemi kullanarak *q*-gözleminin sınıfını belirle. |

**2.2. Lévy Roulette FDB-COA Algoritması**

Çakal Optimizasyon Algoritması (COA), Canis latrans (Çakal) türünden esinlenmiş popülasyon tabanlı bir algoritmadır (Pierezan ve Coelho, 2018). Diğer birçok meta-sezgisel arama algoritmasının aksine, COA'da popülasyon alt gruplara ve sürülere bölünür. Her sürü, belirli sayıda çakal ve bir alfa bireyinden oluşur. Alfa, bir sürünün en iyi çakalını temsil eder. Ayrıca, COA algoritması çakalların sosyal yapılarını ve davranışlarını paylaşmaya odaklanır. Diğer stokastik algoritmalarda olduğu gibi, başlangıçta çakalların sosyal durumları rastgele olarak popülasyon içinde belirlenir. COA, çakallardan gelen bilgilere dayanarak sürünün kültürel eğilimini hesaplar ve çakalların sosyal durumlarını paylaştıklarını ve sürünün sürdürülmesine katkıda bulunmak için düzenlendiğini varsayar. Bu çalışmada, LRFDB-COA (Lévy Roulette Fitness Distance Balance Coyote Optimization Algorithm) adlı algoritma kullanılmıştır. Bu algoritma, COA'nın performansını artırmak için Levy uçuşu ve olasılıksal FDB seçim yöntemini kullanmaktadır. Bu yaklaşıma göre, sürünün kültürel eğilimi, sürüye en fazla katkı yapabilecek sosyal durumlara sahip çakallar tarafından temsil edilebilir.[3]

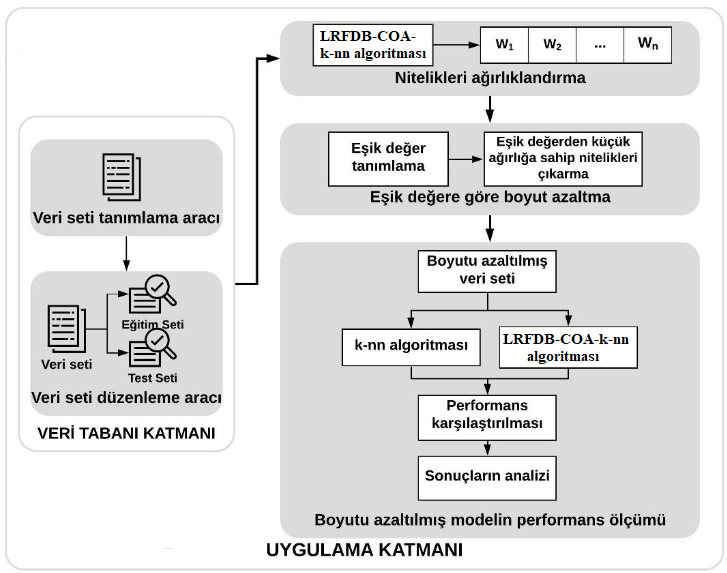
Algoritmanın adımları aşağıda sözde kod şeklinde gösterilmiştir.

**Algoritma 2.** LRFDB-COA algoritmasının temel adımları [3]

|  |
| --- |
| 1. **Başla** |
| 1. *Np* ve *Nc*: Rastgele *Nc* adet çakal barındıran *Np* adet sürü oluştur. |
| 1. **for** Her bir **p** sürüsü için |
| 1. Her bir çakalın uygunluk değerini belirle *fitcp,t : f (soccp,t)* |
| 1. **end for** |
| **Arama Süreci Yaşam Döngüsü** |
| 6. **while** (Maksimum iterasyona ulaşana kadar) |
| **for** Her bir **p** sürüsü için |
| 7. **Rehber Seçimi** |
| 1. Sürüdeki uygunluk değerlerine göre alfa çakalı belirle |
| *alphap,t*= {(*soccp,t***|***argc ={1,2,…Nc}**min f(soccp,t)*} |
| 9. FDB ile sürünün kültürel eğilimini (cultural tendency) hesapla |
| 10. **Komşuluk araması ve Çeşitlilik** |
| 11. **for** Her **p** sürüsündeki her bir **c** çakalı için |
| 1. Çakalın yeni sosyal durumunu hesapla |
| 1. Çakalın yeni uygunluk değerini hesapla *fitcp,t : f (soccp,t)* |
| 1. Çakalın yeni sosyal durumunu güncelle |
| 15. **end for** |
| 16. **Güncelleme Mekanizması** |
| 17. Ölen ve yeni doğan çakallara göre sürüyü güncelle |
| 18. **end for** |
| 1. Sürüler arası transfer |
| 1. Çakalların yaşlarını güncelle |
| 21**. end while** |

**2.3. LRFDB-COA-k-nn Melez Algoritması İle Boyut İndirgeme Çalışması**

Sezgisel boyut indirgeme için geliştirilen LRFDB-COA-k-nnalgoritmasının işleyişi Şekil 1’de verilmiştir. Buna göre sınıflandırma problemlerinde sezgisel boyut indirgeme süreci birkaç adımdan oluşmaktadır. Bu adımlar sırasıyla, probleme ait örnek bir veri setinin tanımlanması, meta-sezgisel arama algoritması kullanılarak niteliklerin ağırlıklandırılması, boyut indirgeme için eşik değerlerin tanımlanması ve indirgeme işlemi sonrası sınıflandırma performansının test edilmesi şeklinde ifade edilebilir. Aşağıda sırasıyla Şekil 1’de verilen öğeler ve süreçler açıklanmaktadır.



Şekil 1

**VERİ TABANI KATMANI**

Veri Tabanı Katmanında ‘Urban Land Cover’ veri seti kullanılmaktadır. İlk olarak veri setine gerekli düzenlemeler yapılmıştır. Yüklenen veri seti belirlenen orana göre eğitim ve test seti olarak ayrılır. Modelin eğitim setiyle k-nn sınıflandırması yapılmakta ve test setiyle başarısı ölçülmektedir. Veri setlerinde toplam örnek sayısının yaklaşık olarak %75’i eğitim, %25’i test için kullanılmıştır.

**UYGULAMA KATMANI**

**LRFDB-COA-k-nn algoritması ile niteliklerin ağırlıklandırılması:** Sezgisel *k*-nn algoritması ile niteliklerin probleme etkisi incelenip buna göre ağırlıklandırılması işlemi yapılmaktadır. Bunun için meta-sezgisel arama algoritmalarından biri olan COA algoritmasının, FDB ve Levy Flight ile iyileştirilmiş hali olan LRFDB-COA kullanılmaktadır. LRFDB-COA algoritmasında çözüm adayları problem niteliklerinin ağırlıklarıdır. Meta-sezgisel arama algoritmasının çözüm adayları 0 ile 1 arasında olacak şekilde kısıtlanmıştır. (Alt ve Üst Sınırlar) Yani ideal ağırlıklar 0 ile 1 arasında aranacaktır. Amaç fonksiyon ise sezgisel *k*-nn fonksiyonudur. Ağırlıkların yani çözüm adayların uygunluk değerlerini ölçmek için amaç fonksiyonundan dönen sınıflandırma hata değerine bakılmaktadır. Sezgisel *k*-nn algoritması için öklid uzaklık bağıntısı Eşitlik 1’de verilmiştir.

(1)

Ağırlıklar *k*-nn algoritmasında kullanılıp *k*-nn’nin sınıflandırma hata değerini minimum yapan ağırlıkların aranması işlemi gerçekleşmektedir. İdeal çözüm adayların aranması işlemi sonlandırma kriteri tamamlanıncaya kadar devam eder.

**Eşik değere göre nitelik seçimi / boyut azaltma:** LRFDB-COA algoritmasının nitelikler için en uygun ağırlıkları arama işleminin sonlandırılmasından sonra problemin boyut azaltma/nitelik çıkarımı aşamasına geçilmektedir. Bu aşamada eşik değer kullanılır. 0 ile 1 arasında bulunan ağırlıklardan, eşik değerden düşük ağırlığa sahip nitelikler çıkarılmaktadır.

**Boyutu azaltılmış modelin performans ölçümü:** Niteliklerin çıkarılması işleminden sonra modelin sınıflandırma performansına bakılır. Klasik *k*-nn algoritması ve sezgisel k-nn algoritmalarının sınıflandırma hata değerlerine bakılmaktadır. Eğer bu performans hata değerleri, nitelik çıkarılmadan önceki modelin sınıflandırma hata değerlerinden düşükse yani sınıflandırma performansı düşmemiş hatta iyileşmişse başarı sağlanmış olacaktır.

**3. Deneysel Çalışmalar**

Bu bölümde uygulamada kullanılan veri setleri, önerilen yöntem kullanılarak veri setlerinde boyut azaltma ve nitelik çıkarımı, çıkarılan bu niteliklerden sonra algoritmanın veri seti üzerindeki sınıflandırma başarısına bakılmaktadır.

**3.1. Ayarlar**

Algoritma ayarları için Tablo 1’de gösterilen değerler aynı şekilde kullanılmıştır.

**Tablo 1.** Algoritmaların Parametre Değerleri

|  |  |
| --- | --- |
| **Algoritma** | **Parametre Değerleri** |
| **LRFDB-COA** | Sürü Sayısı (*Np*) = 20, Çakal Sayısı (*Nc*) = 5, Popülasyon = 100 (*Np\*Nc*), Maksimum İterasyon = 200, Alt Sınır = 0, Üst Sınır = 1 |
| **k-nn** | *k* değeri, Uzaklık Bağıntısı = Öklid |

**3.2. Veri Setleri**

Geliştirilen melez algoritma UCI Machine Learning veri havuzundan temin edilen ‘Urban Land Cover’ veri seti üzerinde tatbik edilmiştir. Veri seti seçilirken sınıflandırma problemine uygun bir veri seti olmasına, makaledeki veri okuma modeline uygun olmasına, veri setinin boyut indirgeme problemine uygun olmasına ve eksik/hatalı veri olmamasına dikkat edilmiştir.

**Tablo 2.** Veri Seti Bilgisi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Boyut** | **Eğitim Örnek** | **Test Örnek** | ***k* [4]** |
| Urban Land Cover | 148 | 507 | 168 | 9 |

**3.3. Deneysel Sonuçların Analizi**

Veri setlerinin boyut azaltılmadan önce LRFDB-COA-k-nn ve k-nn ile sınıflandırma hata değerleri verilmiştir. Tablo 3’te ortalama hata değerlerine bakıldığında sezgisel knn algoritmasının klasik k-nn algoritmasından daha düşük hata değerlerine sahip olduğu yani sınıflandırmada daha başarılı olduğu görülmektedir.

**Tablo 3.** Algoritma Sınıflandırma Hata Değerleri

|  |  |  |
| --- | --- | --- |
| **Algoritma** | | **Urban Land Cover** |
| LRFDB-COA-k-nn | En İyi | 36.90 |
| En Kötü | 47.61 |
| Ortalama | 43.90 |
| Std.Sapma | 2.41 |
| k-nn | Sonuç | 50.59 |

Uygulamanın bir sonraki aşamasında ise boyut azaltma işlemi gerçekleşmiştir. Eşik değerden düşük ağırlığa sahip nitelikler çıkarılmıştır. Eşik değere bağlı olarak çıkarılan nitelik sayısı ve çıkarılan niteliklerden sonra melez algoritmanın sınıflandırma hata değerleri Tablo 4’te verilmiştir.

**Tablo 4.** Boyut Azaltıldıktan Sonra Melez Algoritmanın Sınıflandırma Hata Değerleri

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Eşik Değer |  | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 |
| Çıkarılan Nitelik Sayısı | Ortalama | 32,38 | 43,1 | 53,81 | 64,43 | 76,29 | 87,19 | 99,05 | 109,86 |
| Ortanca | 33 | 42 | 55 | 65 | 77 | 86 | 99 | 111 |
| Std.Sapma | 10,14 | 8,58 | 7,68 | 7,17 | 5,51 | 5,27 | 6,38 | 7,56 |
| k-nn | En İyi | 47,02 | 44,05 | 41,07 | 39,88 | 26,19 | 26,19 | 27,38 | 26,79 |
| En Kötü | 54,17 | 57,14 | 54,17 | 54,17 | 56,55 | 55,95 | 56,55 | 55,95 |
| Ortalama | 51,08 | 51,05 | 50,14 | 48,5 | 47,79 | 46,83 | 48,36 | 45,46 |
| Std.Sapma | 1,9 | 2,93 | 3,36 | 3,91 | 6,44 | 6,87 | 7,27 | 8,09 |
| LRFDB-COA- k-nn | En İyi | 26,79 | 23,81 | 22,02 | 20,83 | 16,07 | 17,26 | 17,26 | 16,07 |
| En Kötü | 44,64 | 47,02 | 42,26 | 41,07 | 39,29 | 39,29 | 43,45 | 33,33 |
| Ortalama | 38,15 | 34,81 | 32,54 | 30,33 | 27,24 | 25,31 | 25,14 | 23,19 |
| Std.Sapma | 4,74 | 5,86 | 5,97 | 5,63 | 6,19 | 5,05 | 6,62 | 4,29 |

A picture containing text, screenshot, diagram, rectangle

Description automatically generated

Şekil 3. Veri Setinde Çıkarılan Niteliklerin Sayısı (%) ve Sınıflandırma Doğruluk Değerleri

Şekil 3 ’te Urban Land Cover veri seti için nitelik çıkarma yapılmadan önce k-nn ve LRFDB-COA-k-nn uygulanarak elde edilen ortalama sınıflandırma doğruluk değerleri, veri setinin ortalama niteliklerinin yüzde kaçının çıkarıldığı ve bu çıkarılan niteliklerden sonra k-nn ve LRFDB-COA-k-nn melez algoritması uygulanarak elde edilen ortalama sınıflandırma doğruluk değerleri verilmiştir.

Şekil 3 ‘e bakıldığında veri setindeki niteliklerin %74.82 ’si çıkarıldığında doğruluk değerlerinin düşmediği hatta her iki algoritmanın performansında iyileşme olduğu görülmüştür.

**4. Sonuç**

Bu çalışmada, LRFDB-COA-k-nn ile sezgisel olarak problem boyutunun azaltılması (nitelik seçimi) ve bu boyut azaltma sonucunda k-nn ve LRFDB-COA-k-nn algoritmalarının sınıflandırma performanslarındaki değişim araştırılmıştır. Probleme ait bağımsız değişkenlerin ağırlıklandırılması ve bu ağırlıklara bakılarak belirlenen eşik değerden düşük niteliklerin problemin bağımsız değişkenleri arasından çıkarılması sağlanmıştır. Problem boyutunun azaltılması, bir başka ifadeyle nitelik seçimi için performans kriteri k-nn/sezgisel k-nn sınıflandırıcının sınıflandırma doğruluğu yüzdesinin boyut azaltmadan önceki performansa göre kötüleşmemesidir. Yani probleme ait bağımsız değişkenlerin sayısı azaltılırken, sınıflandırma algoritmasının performansının en azından aynı kalması şartı aranmıştır. Sonuç olarak Urban Land Cover veri seti üzerinde LRFDB-COA-k-nn algoritması ile yürütülen nitelik seçimi süreci başarılı bir şekilde tamamlanmıştır. Problemlere ait nitelikler çıkarıldıktan sonra sınıflandırma performansının düşmediği hatta bazı veri setlerinde daha da başarılı olduğu görülmüştür. Bu durum birçok problem için modelleme aşamasının hatalı yapıldığına, bağımsız değişkenlerin optimum şekilde belirlenemediğine ve problem modellerinin problemin davranışını yansıtmakta yetersiz olduğuna işaret etmektedir.

**5. Kaynaklar/Referanslar**

1. Çoklu İmge Eşikleme Problemlerinde Metasezgisel Algoritmaların Performans Analizi, Gökhan Pekdemir, Selçuk Üniversitesi Yüksek Lisans Tezi, Şubat 2012.
2. Kahraman, H. T., Sagiroglu, S., & Colak, I. (2013). The development of intuitive knowledge classifier and the modeling of domain dependent data. Knowledge-Based Systems, 37, 283-295
3. Duman, S., Kahraman, H. T., Guvenc, U., Aras, S., (2021). Development of a Levy flight and FDB-based coyote optimization algorithm for global optimization and real-world ACOPF problems. Soft Computing, <https://doi.org/10.1007/s00500-021-05654-z>.
4. Kahraman, H., Aras, B. & Yıldız, O. (2020). SINIFLANDIRMA PROBLEMLERİ İÇİN AGDE-TABANLI META-SEZGİSEL BOYUT İNDİRGEME ALGORİTMASININ GELİŞTİRİLMESİ. Mühendislik Bilimleri ve Tasarım Dergisi, Special Issue: International Conference on Artificial Intelligence and Applied Mathematics in Engineering (ICAIAME 2020) , 206-217 . DOI: 10.21923/jesd.828518