



SINIFLANDIRMA PROBLEMLERİ İÇİN AGDE-TABANLI META-SEZGİSEL BOYUT İNDİRGEME ALGORİTMASININ GELİŞTİRİLMESİ

Hamdi Tolga KAHRAMAN*, Büşra ARAS, Orhun YILDIZ

Karadeniz Teknik Üniversitesi, Teknoloji Fakültesi, Yazılım Mühendisliği Bölümü, Trabzon, Türkiye

Anahtar Kelimeler	Öz
<i>k-nn Sınıflandırma Yöntemi, AGDE, Meta-Sezgisel Ağırlıklandırma, Boyut İndirgeme, Meta-Sezgisel Nitelik Seçimi.</i>	Sınıflandırma problemleri yapay zekanın geniş bir uygulama alanını temsil etmektedirler. Sınıflandırma problemlerinin özelliklerine bağlı olarak algoritmaların performansları değişebilmektedir. Bir sınıflandırma probleminin boyutunun ya da nitelik sayısının değişmesi durumunda performansı bundan etkilenmeyen bir algoritma yoktur. Hemen tüm algoritmalar için problem boyutunun artması performansı olumsuz yönde etkileyen bir faktördür. Dolayısıyla sınıflandırma problemlerinde problem boyutu azaltma üzerinde yoğun bir şekilde çalışılan önemli bir konudur. Özellikle yüksek boyutlu nitelik uzaylarında problem için en uygun modeli yaratmak zor bir görevdir. Karmaşıklık düzeyi yüksek arama uzaylarında genel en iyi çözümü bulmak olanaksız olarak tanımlanmaktadır. Dolayısıyla bu tür arama problemleri için en ideal çözüm yolu meta-sezgisel arama algoritmalarından faydalanmaktır. Bu makale çalışmasında sınıflandırma problemlerinde boyut azaltmak için AGDE (adaptive guided differential evolution) algoritması temelli meta-sezgisel nitelik seçim yöntemi geliştirilmektedir. Geliştirilen yöntem niteliklerin ağırlıklandırılmasını ve eşik değerden küçük ağırlıklı niteliklerin tespit edilmesini sağlamaktadır. Deneysel çalışmalardan elde edilen sonuçlar AGDE-tabanlı nitelik seçim yönteminin sınıflandırma problemleri için etkili bir boyut azaltma aracı olduğunu göstermektedir.

DEVELOPMENT OF AGDE-BASED META-HEURISTIC DIMENSION REDUCTION ALGORITHM FOR CLASSIFICATION PROBLEMS

Keywords	Abstract
<i>k-nn classifier, Adaptive Guided Differential Evolution, Meta-Heuristic Weighting, Dimensionality Reduction, Meta-Heuristic Dimensionality Reduction.</i>	Classification problems represent a wide range of applications of artificial intelligence. Depending on the properties of classification problems, algorithms' performances may vary. If the number of attributes/features of a problem changes, the performance of the algorithm used to solve this problem also changes. For almost all algorithms, increasing problem dimension is a factor that negatively affects performance. Therefore, dimensionality reduction is an important issue that is studied extensively. It is a difficult task to create the most suitable model for the problem, especially in high-dimensional search spaces (with a high number of independent variables). It is impossible to find the best overall solution in highly complex search spaces. Therefore, the ideal solution for such search problems is to use meta-heuristic search algorithms. In this article, meta-heuristic feature selection method based on AGDE (adaptive guided differential evolution) algorithm is developed for dimensionality reduction in classification problems. The developed method enables the weighting of the attributes and the determination of the weighted features less than the threshold value. The results obtained from the experimental studies show that AGDE-based dimensionality reduction method has very effective performance for classification problems.

Alıntı / Cite

Kahraman, H.T., Aras, B., Yıldız, O., (2020). Sınıflandırma Problemleri İçin AGDE-Tabanlı Meta-Sezgisel Boyut İndirgeme Algoritmasının Geliştirilmesi, Mühendislik Bilimleri ve Tasarım Dergisi, 8(5), 206 – 217.

* İlgili yazar / Corresponding author: htolgakahraman@ktu.edu.tr, +90-462-377-8358

Yazar Kimliği / Author ID (ORCID Number)	Makale Süreci / Article Process	
H. T. Kahraman, 0000-0001-9985-6324	Başvuru Tarihi / Submission Date	19.11.2020
B. Aras, 0000-0001-9214-771X	Revizyon Tarihi / Revision Date	19.12.2020
O. Yıldız, 0000-0002-6191-776X	Kabul Tarihi / Accepted Date	19.12.2020
	Yayın Tarihi / Published Date	29.12.2020

1. Giriş (Introduction)

Nitelik sayısı fazla ve karmaşıklık düzeyi yüksek yapay zekâ problemleri için etkili çözümler geliştirmek zordur. Bu tür zor problemler için bir yapay zekâ çözümü geliştirmeden önce problem modelinin en etkili şekilde oluşturulmasına ihtiyaç vardır. Bunun için problemi tanımlayan niteliklerin (bağımsız değişkenlerin) hedef parametre (bağımlı değişken) üzerindeki etkilerinin belirlenmesine (ağırlıklandırılmasına) ihtiyaç vardır. Hedef parametre üzerinde yeterince etkiye sahip olmayan nitelikler belirlendikten sonra bunların problem modelinden çıkartılması mümkün olabilir. Bu süreç problemin boyutunun azaltılması ve problemi modellemek için en uygun niteliklerin belirlenmesi olarak ifade edilebilir. Bu süreçte problem niteliklerinin ağırlıklandırılması için kestirim yapılmasına yani optimizasyon çalışmasına ihtiyaç vardır. Optimizasyon çalışmaları neticesinde, problem için kabul edilebilir bir sınıflandırma performansı sağlayan ve asgari sayıda nitelikten oluşan bir modelin yaratılması sağlanabilir. Optimizasyon sürecinde ise problem boyutunun ve karmaşıklık düzeyinin fazla olması, meta-sezgisel arama algoritmalarının kullanılması gerekliliğini ortaya koymaktadır.

Meta-sezgisel arama (MSA) algoritmaları, yüksek karmaşıklığa sahip optimizasyon/kestirim çalışmalarında en etkili yöntemlerdir. MSA algoritmaları kullanılarak farklı alanlardan binlerce optimizasyon çalışması yürütülmüştür. MSA'lar sadece optimizasyon sürecinde arama algoritmaları olarak değil birçok çalışmada daha güçlü ve etkili melez yapay zekâ algoritmaları tasarlamak için de kullanılmaktadırlar. Bu amaçla tahmin (Kahraman vd., 2012), sınıflandırma (Kahraman, 2016; Kahraman vd., 2013; Yılmaz vd., 2018) ve kümeleme (Cerrada vd., 2019; Li vd., 2020; Yaday ve Prakash, 2020; Zhou vd., 2019; Aliarah vd., 2020) ve optimizasyon problemlerinin (Özkaraca ve Keçebaş., 2019; Özkaraca vd., 2018; Özkaraca vd. 2017) çözümlenmesi için geliştirilmiş MSA-tabanlı melez algoritmalar mevcuttur. MSA'ları doğadan esinlenilerek geliştirilmiş tekniklerdir. 1950'li yıllardan bu yana çok sayıda MSA algoritması geliştirilmiştir (Booker vd., 1989; Ramos-Figueroa vd., 2020; Kulkarni vd., 2019). Son on yılda ise bu sayı her yıl onlarca yeni MSA algoritmasının geliştirilmesi ile artmaktadır. Geliştirilen yüzlerce algoritma arasından en etkili olanı belirlemek ise zor bir iştir. Çünkü tüm problemlerde en iyi olan bir MSA tekniği yoktur. MSA'ların performansları problem bağımlıdır. Bu doğada da böyledir. 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 olmaktadır. Bu makale çalışmasında optimizasyon sürecinde kullanılan arama algoritması da oldukça yeni ve etkili bir MSA tekniği olan Adaptive Guided Differential Evaluation (AGDE) dir (Lin vd., 2016). AGDE, adından da anlaşılacağı üzere Differential Evaluation (DE)'nin (Yusta, 2009) performansı iyileştirilmiş bir varyasyonu olarak geliştirilmiş algoritmadır. DE'nin sahip olduğu güçlü temeller AGDE'nin de başarısındaki en büyük etkenlerdir.

Bu makale çalışmasında birçok farklı meta-sezgisel yöntem kullanılarak gerçekleştirilmiş olan (Tran vd., 2018; Dash ve Liu, 1997; Tang vd., 2014; Xue vd., 2012) ve literatürde boyut indirgeme (dimensionality reduction) olarak bilinen konu üzerine çalışılmaktadır. Bu amaçla melez bir yapay zekâ algoritması geliştirilmiş ve problem uzayı bağımsız değişkenleri sayısal veri tipinde olan sınıflandırma problemleri olarak belirlenmiştir. Amaç bu tipteki problemlere ait veri setlerini kullanarak sınıflandırma performansını korurken nitelik sayısını azaltmaktır (boyut indirgeme). Sınıflandırma algoritması olarak k-en yakın komşu (k-nearest neighbor, k-nn) ve sezgisel k-nn algoritmaları kullanılmıştır. MSA algoritması olarak ise güncel ve güçlü bir MSA algoritması olan AGDE kullanılmıştır. Bu amaçla geliştirilen boyut indirgeme algoritması bir optimizasyon yöntemi olmanın da ötesinde birden fazla algoritmanın bir arada kullanıldığı melez bir yöntemdir. Bu melez algoritma, k-nn sınıflandırıcı için en iyi k-değerinin belirlenmesi, sınıflandırma eşik değerinin tanımlanması, probleme ait niteliklerin ağırlıklandırılması ve sezgisel sınıflandırma gibi farklı gereksinimlere cevap veren yeteneklere sahip olarak geliştirilmiştir. Makale için ihtiyaç duyulan sınıflandırma problemlerine ait veri setleri için de UCI Machine Learning veri havuzu kullanılmıştır (Alizadehsani vd., 2013; Johnson, 2013).

Makale bölümleri sırasıyla, yöntem, deneysel çalışma ve sonuçlar şeklinde tasarlanmıştır. Yöntem bölümünde öncelikle, bu makalede geliştirilen melez boyut indirgeme algoritmasının temel öğeleri olan k-nn sınıflandırıcı ve AGDE algoritması tanıtılmaktadır. Daha sonra önerilen yöntem adım-adım açıklanmıştır. Deneysel çalışma bölümünde ise dört farklı probleme ait veri setleri kullanılarak geliştirilen melez algoritmanın bu problemler üzerindeki performansı test edilmiştir. Son olarak bu çalışmadan elde edilen sonuçlar değerlendirilmiştir.

2. Materyal ve Yöntem (Material and Method)

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

2.1. k -En Yakın Komşu Algoritması (k -Nearest Neighbour Algorithm)

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 sınıflandırma sürecinin ilk adımlarından biri k -değerinin belirlenmesidir. k değeri için en uygun değer belirlenmesi zor bir görevdir. Çünkü k -değerinin belirlenmesinde kullanılan bir matematiksel bir eşitlik bulunmamaktadır. Üstelik k -değeri probleme bağlı olarak değiştiği kadar örnek gözlemleri içeren veri setindeki gözlem sayısına ve veri setinin oluşturulmasında izlenen yola bağlı olarak değişir. Algoritmanın performansı üzerinde etkili olan başka öğe de gözlemler arasındaki uzaklıkların hesaplanmasında kullanılan uzaklık metrikleridir. Hızlı ve etkili oldukları için bu süreçte Öklit, Manhattan ve Minkowski gibi uzaklık bağıntıları sıklıkla kullanılmaktadır. Bunun yanında bulanık uzaklık metriği klasik metriklerle kıyasla çok daha etkili sonuçlar vermektedir. Bulanık uzaklık metriğinin ortalama olarak k -nn algoritmasının sınıflandırma performansını klasik metriklerle göre %10'dan fazla iyileştirdiği bilinmektedir (Kahraman, 2016; Kahraman vd., 2013; Yılmaz vd., 2018).

Algoritma 1. k -nn Algoritmasının Sözde Kodu (Kahraman vd., 2013) (Pseudo Code of the k -nn Algorithm)

1. *Veri setinin tanımlanması:* probleme ait n -adet örnek gözlemleri içeren ve problem uzayını temsil etme kabiliyeti yüksek (gözlem uzayını homojen olarak örnekleyen) \mathbf{X} veri setini oluştur.
 2. *Uzaklık bağıntısının belirlenmesi:* gözlemler arasındaki uzaklıkların hesaplanmasında kullanılacak yöntemi belirle.
 3. *k -değerinin belirlenmesi:* gözlem sayısına ve veri setinin karakteristiğine bağlı olarak k -komşu sayısı için arama uzayının sınırlarını tanımla.
 - for each** k_j
 - k_j için sınıflandırma performansını $SPk_j = f_{k\text{-nn}}(k_j)$
 - if** ($SPk_j > SPk_{j-1}$)
 - 4. $k = k_j$
 - end if**
 - end**
 5. En iyi sınıflandırma performansı sağlayan k -değerini kaydet
 6. Sınıf etiketi belirlenecek olan q sorgu gözlemini tanımla
 - for** $i=1:n$
 - 7. $D_{[i]}=q$ ile \mathbf{X}_i arasındaki uzaklığı hesapla
 - end**
 - 8. $X_{q[k]}=D_{[i]}$ uzaklık dizisinden q sorgu gözlemine en yakın k -adet gözlemi belirle
 - 9. $X_{q[k]}$ gözlemlerinin sınıflarını dikkate alarak çoğunluk oylaması/ağırlıklı oylama yöntemi kullanarak q -gözleminin sınıfını belirle.
-

Klasik k -nn algoritmasında gözlemler arasındaki uzaklık hesabı öklit bağıntısı kullanılarak Eşitlik (1)'de verildiği gibi hesaplanır (Kahraman vd., 2013).

$$d(x_1, x_2) = \sqrt{(x_{11} - x_{21})^2 + (x_{12} - x_{22})^2 + \dots + (x_{1n} - x_{2n})^2} \quad (1)$$

Eşitlik (1)'de sınıflandırma probleminin nitelik sayısı n -ile, X_1 ve X_2 gözlemleri ise $\langle x_{11}, x_{12}, x_{13}, \dots, x_{1n} \rangle$ ve $\langle x_{21}, x_{22}, x_{23}, \dots, x_{2n} \rangle$ ile temsil edilmektedir. Buna göre uzaklık hesabında problemin niteliklerinin eşit derecede etkiye sahip olduğu görülmektedir. Bunun yanında her bir niteliğin alt ve üst sınır değerlerine bağlı olarak sayısal büyüklüklerinin de uzaklık hesabı üzerinde farklı bir etki yaratabileceği dikkate alınmalıdır. Bu durum niteliklerin birbirlerine baskınlık kurması olarak nitelendirilir. Baskınlık durumunu bertaraf etmek için her bir niteliğin normalizasyonu $[0, 1]$ arasında ölçeklendirilmesi mümkündür. Ancak bu durumda ise probleme ait niteliklerin, gözlemlerin sınıflandırılmasında eşit derecede etkiye sahip olmaları gibi bir durum ortaya çıkar. Örneğin bir müşteriye verilecek banka kredisi hesaplanırken kişinin mesleği, aylık geliri, kredi skoru, borcu, cinsiyeti, yaşı,

yaşadığı şehir, varlıkları gibi bilgiler dikkate alınabilir. Ancak bu niteliklerin her biri kişiye verilecek kredinin miktarının belirlenmesinde farklı derecede öneme sahiptir.

Dolayısıyla klasik k-nn algoritmasının işleyişi incelendiğinde, sınıflandırma sürecinde niteliklerin etkilerinin eşit olduğunun kabul edilmesi doğru ve etkili bir yaklaşım değildir. Bu nedenle niteliklerin önem/etki derecelerinin belirlenmesi için sezgisel arama algoritmalarının kullanıldığı yöntemler geliştirilmiştir. En iyi sınıflandırma performansının elde edildiği ağırlık değerlerinin arandığı bu yöntemler literatürde oldukça yaygın bir şekilde kullanılmaktadırlar. Takip eden bölümde, sezgisel sınıflandırma algoritmasının iki temel ögesinden biri olan meta-sezgisel arama algoritmalarının güncel bir örneği tanıtılmaktadır.

2.2. AGDE Sezgisel Arama Algoritması (AGDE Heuristic Search Algorithm)

AGDE algoritması, popülasyon tabanlı sezgisel bir optimizasyon tekniği olmakla birlikte global çözümü bulma açısından basit ama aynı zamanda da güçlü bir tekniktir (Lin vd., 2016). AGDE algoritmasının temeli DE (differential evolution) algoritmasına dayanmaktadır. DE algoritmasının (Yusta, 2009) temeli ise genetik algoritmaya (Booker vd., 1989) dayanır. Algoritmanın adımları Algoritma 2’de verilmiştir.

Algoritma 2. AGDE Algoritmasının Temel Adımları (Lin vd., 2016) (Basic Steps of AGDE Algorithm)

- i) Problemin yaratılması (uygunluk fonksiyonunun, ceza fonksiyonunun tanımlanması)
 - ii) Çözüm adayının tasarımı ve çözüm adayları topluluğunun yaratılması
 - iii) Çözüm adayların uygunluk değerlerinin hesaplanması
 - iv) İteratif süreç (*sonlandırma kriteri sağlanıncaya kadar devam et: amaç fonksiyonu azami değerlendirme sayısı*)
 - Seçim süreci: Popülasyondan $r_1 \neq r_2 \neq r_3 \neq i$ olmak üzere üç çözüm adayını rastgele seç
 - Mutasyon
 - Çaprazlama
 - v) Sonlandırma kriteri sağlandı mı?
 - Hayır (Adım iv’e dön)
 - Evet (Arama sürecini sonlandır ve en iyi çözüm adayını kaydet)
-

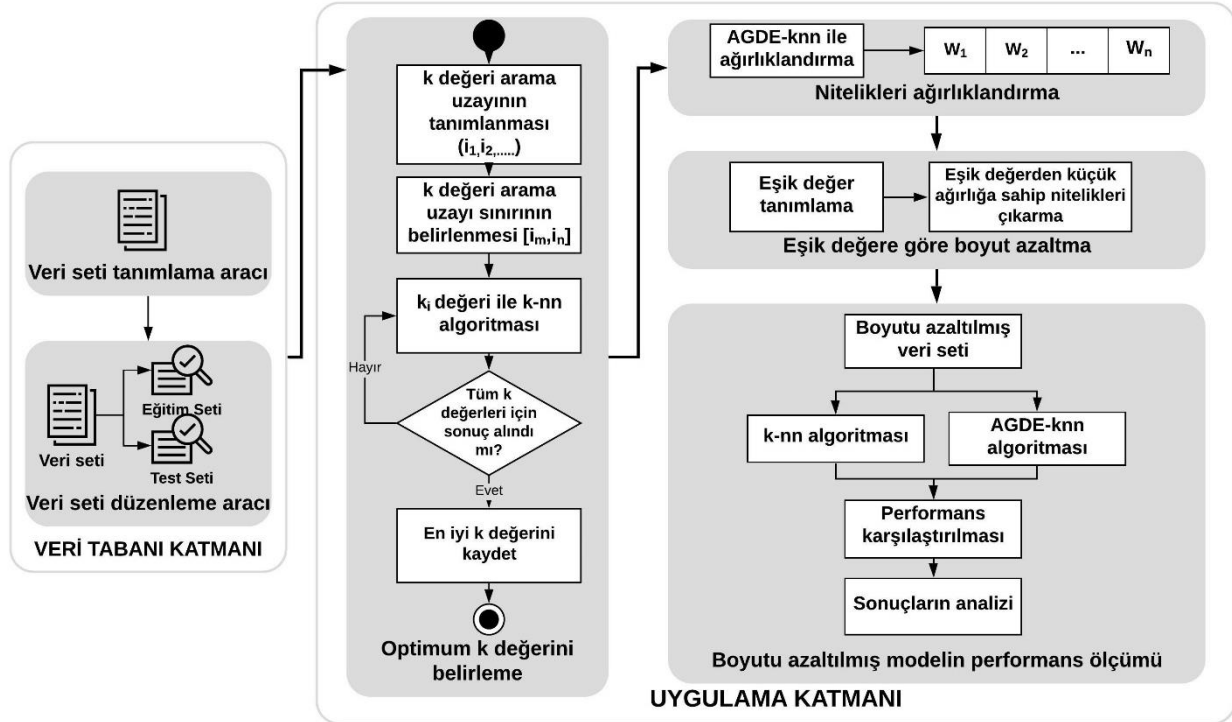
Algoritma 2’de verilen bilgilere göre DE algoritmasında arama süreci yaşam döngüsü seçim işlemi, çaprazlama ve mutasyon olmak üzere üç adımdan oluşur. AGDE algoritmasında da bu üç adım ortaktır fakat AGDE algoritmasının geliştirilmesinde çeşitlilik yeteneğinin iyileştirilmesine yönelik olarak yeni bir mutasyon tekniği önerilmiştir. Bu amaçla Algoritma 2’de seçim sürecinde tanımlanan r_1 , r_2 ve r_3 çözüm adaylarının seçilme yöntemleri DE algoritmasından farklı şekilde uygulanmaktadır. Bu üç çözüm adayının seçilme süreci ve AGDE algoritmasında işlevleri hakkında detaylı bilgi almak için referans çalışma incelenebilir (Lin vd., 2016). AGDE algoritmasında çalışılan ikinci konu ise arama sürecinde komşuluk araması ve çeşitlilik arasındaki dengenin sağlanması için yeni bir adaptasyon şemasının önerilmesidir. AGDE’nin arama performansı mutasyon ve çaprazlama operatörlerinin başarısına bağlıdır. Çaprazlama sürecinin etkisi ise çaprazlama oranına (CR) bağlıdır. CR’nin büyük olması arama sürecinde çeşitliliğe katkı sağlarken hassas aramayı ise engellemektedir. Bu durum genetik algoritma ve diferansiyel evrim algoritmasında da aynıdır. CR’nin küçük olması halinde ise komşuluk araması hassas bir şekilde yerine getirilmekte ancak bu defa da yerel çözüm tuzaklarına yakınsama problemi ortaya çıkabilmektedir. Bu nedenlerden ötürü, AGDE algoritmasında CR parametresinin problem tipine ve arama sürecinin gereksinimlerine bağlı olarak dinamik bir şekilde ayarlanması ihtiyacına yönelik çözüm geliştirilmiştir. Eşitlik (2)’de çaprazlama oranının dinamik bir hale getirilmesi için jenerasyon sayısına bağlı olarak geliştirilmiş ve rastgeleliği içeren bir yöntem verilmektedir. AGDE algoritmasında çaprazlama oranı CR değerlerini üretmek için, önceden belirlenmiş bir aday havuzu kullanır. Bu havuzlar $CR_1 \in [0.05, 0.15]$; $CR_2 \in [0.9, 0.1]$.

$$\begin{cases} \text{if } G = 1, CR_i^1 = \begin{cases} CR_1, \text{ if } rand(0,1) \leq 0.5 \\ CR_2, \text{ else} \end{cases} \\ \text{else } , CR_i^G = \begin{cases} CR_1, \text{ if } rand(0,1) \leq p_1 \\ CR_2, \text{ if } p_1 \leq rand(0,1) \leq p_1 + p_2 \end{cases} \end{cases} \quad (2)$$

Eşitlik (2)’de verilen $rand(0,1)$ ifadesi uniform dağılıma sahip random bir değer üretmek için ve p_1 ve p_2 ise çaprazlama oranı havuzundaki herhangi bir CR setinin seçilme olasılıklarını göstermektedir. Sonuç olarak AGDE algoritması geleneksel DE yönteminden farklı olarak çaprazlama oranının dinamik bir şekilde ayarlandığı yeni bir yöntemi uygulayarak daha etkili bir arama performansı sergilemektedir. Daha fazla bilgi edinmek için lütfen referans çalışmayı inceleyiniz (Lin vd., 2016).

2.3. Önerilen Yöntem: AGDE-knn ile Sezgisel Boyut İndirgeme (Proposed Method: Heuristic Dimension Reduction with AGDE-knn)

Sezgisel boyut indirgeme için geliştirilen AGDE-knn algoritmasının işleyişi Şekil 1’de verilmektedir. Buna göre sınıflandırma problemlerinde sezgisel boyut indirgeme süreci birkaç adımdan oluşmaktadır. Bu adımlar sırasıyla, probleme ait örnek bir veri setinin tanımlanması, veri seti için en uygun k -değerinin belirlenmesi, sezgisel sınıflandırma 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. AGDE-knn ile Sezgisel Boyut İndirgeme Sürecinin Öğeleri (Elements of the Heuristic Dimension Reduction Process with AGDE-knn)

Veri Tabanı Katmanı

Makalede dört farklı boyutta veri seti kullanılmaktadır. İlk olarak veri setleri incelenmekte, gerekli düzenlemeler yapılmaktadır. Yüklenen veri seti iki ayrı alt veri setine bölünmüştür: Eğitim seti ve test seti. Modelin eğitim setiyle eğitilmesi sağlanmakta ve test setiyle başarısı ölçülmektedir. Veri setlerinde toplam örnek sayısının yaklaşık olarak %70'i eğitim, %30'u test için kullanılmıştır.

Uygulama Katmanı

Optimum k değerini belirleme: İlk olarak her bir veri seti k -nn algoritmasıyla tatbik edilmektedir. Klasik k -en yakın komşu algoritmasında niteliklerin problem üzerindeki etkileri aynı kabul edilmektedir. Optimum k -değerini belirleme modülünün sözde kodu Algoritma 3'te verilmiştir.

Algoritma 3. Optimum k -değeri Belirlemenin Sözde Kodu (The Pseudo Code for Determining the Optimum k -parameter)

1. **Başla**
2. k -değeri arama uzayı tanımlanır (k_1, k_2, \dots) .
3. k -değeri arama uzayının alt ve üst sınırı tanımlanır $[m, n]$.
4. **for** $i = m:n$
5. k_i - değeri ile k -nn sınıflandırıcı çalıştırılır.
6. **H:** Bulunan sınıflandırma hata değerleri kaydedilir.
7. **end for**
8. $sayac = 6$ olarak tanımlanır.
9. **while** ($sayac > 0$)
10. **for** $i = m:n$

```

11.         if (Hi < Hi+1)
12.             enKucukHata = Hi olarak seçilir.
13.             sayac bir azaltılır (sayac = sayac - 1).
14.         else
15.             sayac bir artırılır (sayac = sayac + 1).
16.         end if
17.     end for
18. end while
19. enKucukHata değeri elde edilirken kullanılan k değeri seçilir.
20. Bitir

```

k-nn algoritmasıyla niteliklerin henüz çıkarılmadığı, boyut azaltmanın henüz gerçekleşmediği klasik modelin sınıflandırma hata değerleri elde edilmektedir. Her bir veri setinde ilk olarak k değeri arama uzayı tanımlanır. Daha sonra k değeri arama uzayının alt ve üst sınırları tanımlanır. Alt sınırdan başlanarak k-değeri kullanılarak elde edilen hata değeri ile k-değerleri artırılıp bulunan hata değerleri birbiriyle karşılaştırılmaktadır. Bu döngü, bulunan en düşük hata değerinden sonraki 6 iterasyonda hata değerinde azalma olmadığı sürece devam etmektedir. Yani elde edilen hata değerinden sonraki 6 iterasyonda k-nn sınıflandırıcının performansı iyileşmediğinde, o hata değeri elde edilirken kullanılan k-değeri optimum k değeri olarak belirlenmektedir. Her bir veri seti için k-nn algoritmasının ideal k-değeri belirlendikten sonra niteliklerin ağırlıklandırılması işlemine geçilmektedir.

AGDE-knn 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 AGDE algoritması kullanılmaktadır. AGDE algoritmasında çözüm adayları problem niteliklerinin ağırlıklarıdır. Meta-sezgisel arama algoritmasının çözüm adayları 0 ile 1 arasında olacak şekilde kısıtlanmıştır. Yani ideal ağırlıklar 0 ile 1 arasında aranacaktır. Amaç fonksiyon ise sezgisel k-nn'dir. Ağırlıkların yani çözüm adayların uygunluk değerlerini ölçmek için hedef (amaç) fonksiyonundan dönen sınıflandırma hata değerine bakılmaktadır. Sezgisel k-nn algoritması için öklit uzaklık bağıntısı Eşitlik (3)'te verilmiştir.

$$d(x_1, x_2) = \sqrt{w_1(x_{11} - x_{21})^2 + w_2(x_{12} - x_{22})^2 + \dots + w_n(x_{1n} - x_{2n})^2} \quad (3)$$

Algoritma 4. AGDE-knn Algoritması ile Niteliklerin Ağırlıklandırılmasının Sözde Kodu (Pseudo Code of Weighting Attributes with the AGDE-knn Algorithm)

```

1. Başla
2.   P: Problemin n nitelikli ağırlık dizisini temsil eden çözüm adaylarından rastgele bir popülasyon oluşturulur
3.   for i = 1:n
4.       F: Her bir çözüm adayının ağırlık dizisini sezgisel k-nn'e göndererek uygunluk değeri (hata değeri) hesaplanır
5.   end for
6.   //Arama süreci yaşam döngüsünün başlangıcı
7.   while (G1'den Gmax'a (maksimum uygunluk değerine) kadar git)
8.       for i = 1:n
9.           Rastgele mutasyon faktörü üretilir
10.          Üç tane vektör seçilir: bir tane rastgele (XrG), bir tane en iyi çözüm adayı vektörü (Xp_enIyiG), bir tane en kötü çözüm adayı vektörü (Xp_enKötüG)
11.          D: Mutasyon noktası rastgele belirlenir
12.          for j = 1:D
13.              Çaprazlama yapılır
14.              yeniÇözüm: Yeni çözüm adayı elde edilir
15.          end for
16.          if (F(yeniÇözüm) <= F(Xp_enIyiG))
17.              Xp_enIyiG = yeniÇözüm
18.          else
19.              XiG+1'e çaprazlama işlemi yapılır
20.          end if
21.        end for
22.        Sonlandırma kriterine kadar yeni jenerasyon oluştur
23. Bitir

```

Yani ağırlıklar k -nn algoritmasında kullanılıp k -nn'nin sınıflandırma hata değerini minimum yapan ağırlıkların aranması işlemi gerçekleştirilmektedir. İdeal çözüm adayların aranması işlemi sonlandırma kriteri tamamlanıncaya kadar devam eder.

Eşik değere göre nitelik seçimi / boyut azaltma: AGDE algoritmasının nitelikler için en uygun ağırlıkları arama işleminin sonlandırılmasından sonra problemin boyut azaltma/nitelik çıkarımı aşamasına geçilmektedir. Bu aşamada eşik değeri 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 Sonuçlar (Experimental Results)

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

3.1. Ayarlar (Settings)

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

Tablo 1. Algoritmaların Parametre Değerleri (Parameter Values of Algorithms)

Algoritma	Parametre Değerleri
AGDE	$N_p = 50$, $\maxIteration = 300$, $GEN = \text{ceil}(\maxIteration / N_p)$
k -NN	Uzaklık Bağlantısı = Öklid

3.2. Veri Setleri (Datasets)

Makalede geliştirilen algoritma UCI Machine Learning veri havuzundan temin edilen dört farklı veri seti üzerinde tatbik edilmiştir. Veri seti seçilirken sınıflandırma problemine uygun bir veri seti olmasına, makaledeki veri okuma modeline uygun olmasına, veri setlerinin farklı boyutlarda olmasına ve eksik/hatalı veri olmamasına dikkat edilmiştir. Veri okuma modeline uymayan veri setleri üzerinde gerekli değiştirmeler yapılarak, veri setinin geliştirilen algoritma için sağlıklı çalışması sağlanmıştır.

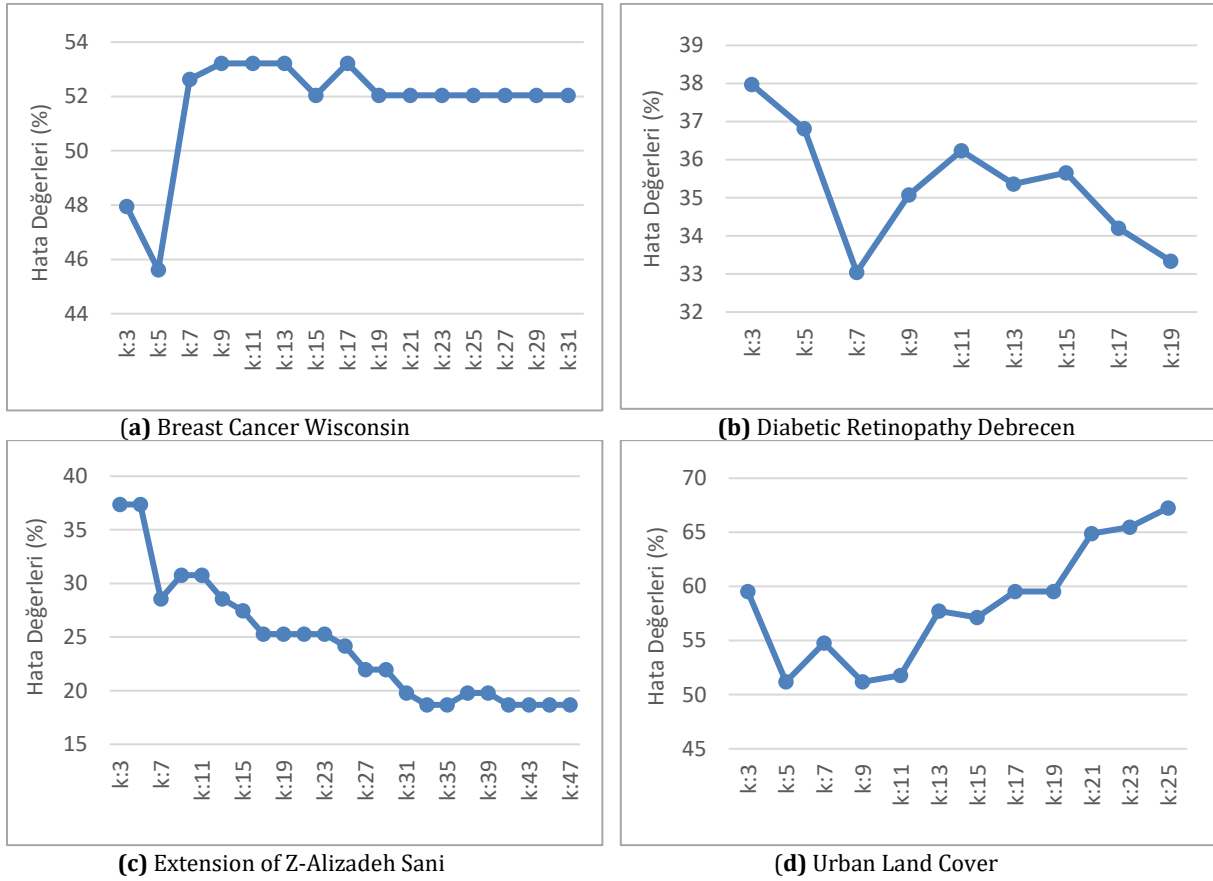
Tablo 2. Veri Setleri Bilgileri (Datasets Information)

	Breast Cancer Wisconsin	Diabetic Retinopathy Debrecen	Extension of Z-Alizadeh Sani	Urban Land Cover
Boyut	32	20	56	148
Eğitim Örnek	398	805	213	392
Test Örnek	171	345	91	168

Kullanılan veri setlerindeki bağımlı değişken sayısı 1 olmakla birlikte bağımsız değişken sayısı sırasıyla 19, 31, 55 ve 147 şeklinde değişiklik göstermektedir.

3.3. Deneysel Sonuçların Analizi (Analysis of Experimental Results)

Uygulamada ilk olarak k -nn algoritması tatbik edilip ideal k -parametresinin tespit edilmesi üzerine çalışmalar yapılmıştır.



Şekil 2. Veri Setlerinin k -nn Algoritmasıyla Sınıflandırma Hata Değerleri (Classification Error Rate of Datasets with k -nn Algorithm)

Şekil 2-a'ya göre veri seti için k komşu sayısının 5 olarak alındığında k -nn algoritmasının sınıflandırma hata değerinin en az olduğu görülmektedir. Şekil 2-b'ye göre veri seti için k -değeri 7 olarak alındığında k -nn algoritması sınıflandırma hata değerinin en düşük geldiği görülmektedir. Şekil 2-c'de ise veri setinin k 'nin 33 ve 35 olarak alındığında k -nn algoritması ile sınıflandırma hata değerinin daha düşük geldiği görülmektedir. Bu veri seti için k 33 olarak alınmıştır. Şekil 2-d'ye göre veri seti için k komşu sayısının 5 ve 9 olarak alındığı durumlarda k -nn algoritmasının sınıflandırma hata değerinin en az olduğu görülmektedir. Veri seti için k 5 olarak alınmıştır.

Veri setleri için belirlenen k değerleri uygulamanın devamında da kullanılmıştır. Tablo 3'te veri setlerinin boyut azaltılmadan önce AGDE-knn 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.

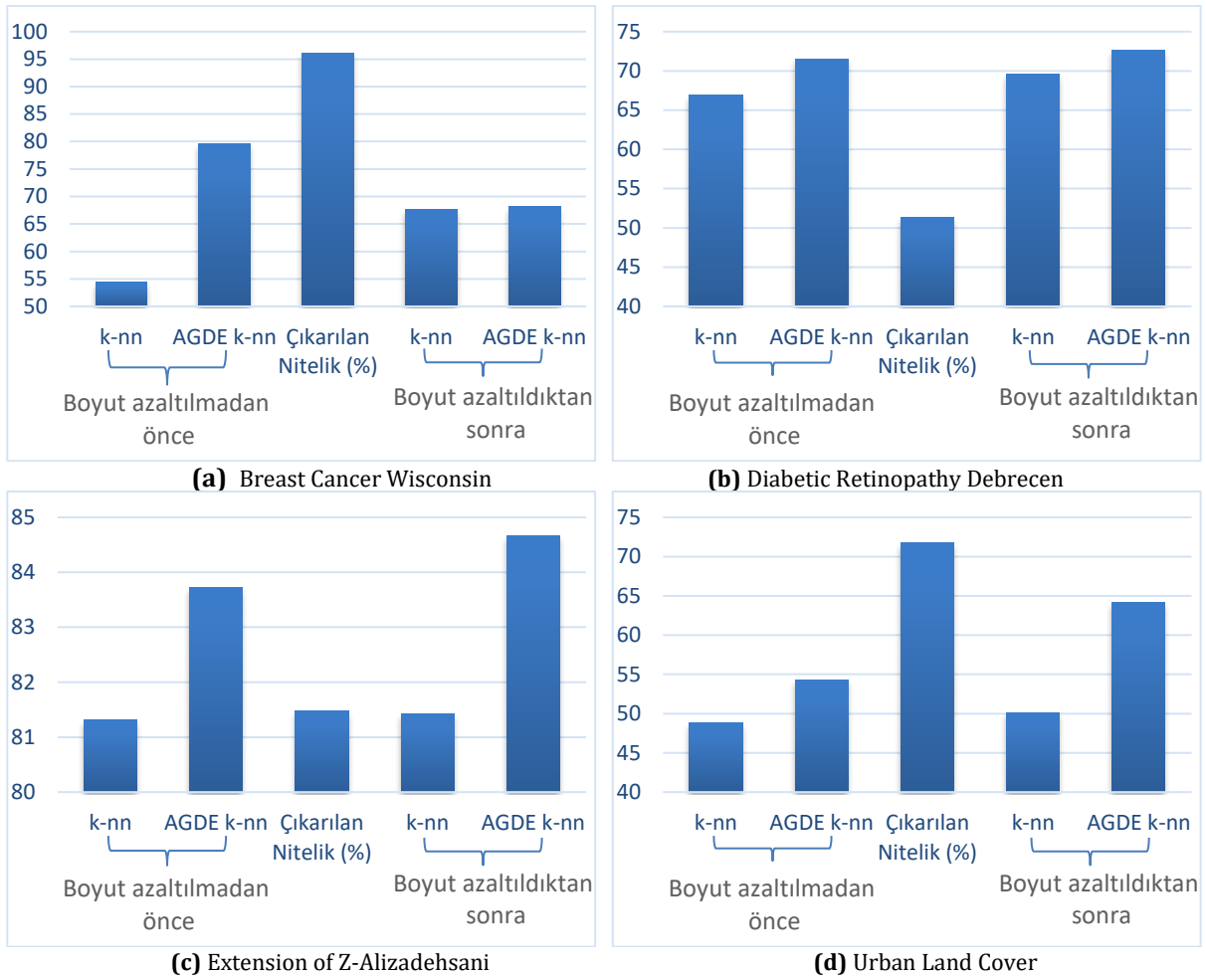
Tablo 3. Algoritmaların Sınıflandırma Hata Değerleri (Classification Error Rate of Algorithms)

Algoritma	Breast Cancer Wisconsin	Diabetic Retinopathy Debrecen	Extension of Z-Alizadehsani	Urban Land Cover	
AGDE- k NN	En İyi	19.88	26.67	42.26	
	En Kötü	34.50	29.57	47.62	
	Ortalama	24.43	28.22	16.06	45.70
	Std. Sapma	4.47	0.69	0.62	0.98
k NN	Sonuç	52.63	33.04	18.68	51.19

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

Tablo 4. Boyut Azaltıldıktan Sonra Algoritmaların Sınıflandırma Hata Değerleri (Classification Error Rate of Algorithms After Dimension Reduction)

Eşik Değer		0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	
DIABETIC RETINOPATHY DEBRECEN	Çıkarılan Nitelik Sayısı	Ortalama	2.81	5.05	6.86	9.14	9.76	12.29	13.95	15.38
		Ortanca	3	5	7	9	9	12	14	16
		Std.Sapma	1.12	1.75	1.24	1.82	2.21	1.85	1.20	1.83
	k-NN	En İyi	30.14	28.41	28.98	26.67	26.96	26.96	27.54	29.86
		En Kötü	35.07	33.91	33.62	39.42	33.91	48.70	40.29	52.17
		Ortalama	31.70	30.75	31.01	31.03	30.37	33.83	33.83	37.65
		Std.Sapma	1.16	1.54	1.60	2.48	1.83	5.08	3.58	5.76
	AGDE-kNN	En İyi	25.22	25.80	25.22	25.22	24.93	25.22	25.80	27.25
		En Kötü	28.12	28.99	29.86	32.17	30.14	42.32	36.81	52.17
		Ortalama	26.64	26.79	27.30	27.38	27.34	30.56	30.88	35.00
		Std.Sapma	0.85	0.71	1.18	1.53	1.38	4.45	3.16	6.65
	BREAST CANCER WISCONSIN	Çıkarılan Nitelik Sayısı	Ortalama	3.48	5.95	9.62	11.95	15.67	18.38	22.38
Ortanca			4	6	9	12	16	19	22	25
Std.Sapma			1.40	2.04	2.40	1.96	2.18	2.44	2.42	1.95
k-NN		En İyi	13.45	13.45	13.45	11.70	11.70	12.87	11.11	11.11
		En Kötü	15.79	15.79	16.38	20.47	19.88	20.47	43.27	36.26
		Ortalama	13.78	13.81	13.98	15.01	15.54	15.23	17.49	17.46
		Std.Sapma	0.84	0.91	1.01	2.24	2.31	2.48	6.80	5.07
AGDE-kNN		En İyi	9.94	11.11	9.36	9.94	9.94	9.36	9.36	7.60
		En Kötü	12.28	12.87	12.28	18.13	14.62	15.79	28.65	31.58
		Ortalama	12.03	12.09	11.92	12.20	12.28	11.97	14.05	13.98
		Std.Sapma	0.63	0.43	0.70	1.64	1.27	1.55	4.26	4.92
EXTENSION OF Z-ALIZADEHSANI		Çıkarılan Nitelik Sayısı	Ortalama	3.71	9.67	14.14	20.24	27.90	32.38	39.95
	Ortanca		4	9	14	21	28	32	40	44
	Std.Sapma		1.74	2.90	3.60	3.95	4.38	3.94	3.19	3.16
	k-NN	En İyi	15.38	15.38	14.29	15.38	15.38	15.38	15.38	14.29
		En Kötü	18.68	18.68	20.88	20.88	23.08	20.88	25.28	21.98
		Ortalama	18.52	18.26	18.31	17.95	19.00	18.26	19.57	18.58
		Std.Sapma	0.72	1.07	1.64	1.78	2.26	1.79	2.27	1.77
	AGDE-kNN	En İyi	13.19	10.99	12.09	10.99	10.99	10.99	12.09	8.79
		En Kötü	17.58	17.58	17.58	16.48	16.48	16.48	17.58	20.88
		Ortalama	15.91	15.38	15.07	13.55	14.02	13.61	15.28	15.33
		Std.Sapma	0.96	1.70	1.67	1.57	1.55	1.32	1.55	2.92
	URBAN LAND COVER	Çıkarılan Nitelik Sayısı	Ortalama	12.00	25.95	42.19	56.10	73.24	92.33	105.48
Ortanca			10	25	41	56	72	93	107	122
Std.Sapma			5.07	4.26	5.01	5.44	7.06	6.90	6.33	4.70
k-NN		En İyi	48.81	48.81	47.62	47.02	50.00	44.64	29.17	18.45
		En Kötü	54.17	54.17	54.76	55.36	57.74	57.74	58.33	64.88
		Ortalama	51.59	50.88	51.73	52.47	53.00	53.06	49.86	52.10
		Std.Sapma	1.59	1.54	1.97	2.13	2.04	3.76	7.54	10.51
AGDE-kNN		En İyi	42.86	41.07	41.07	34.52	32.74	29.17	17.86	16.07
		En Kötü	47.02	48.21	48.21	47.02	48.21	47.62	47.62	50.00
		Ortalama	44.81	44.27	44.30	42.80	42.63	39.17	35.83	36.34
		Std.Sapma	1.19	1.44	1.79	3.28	3.85	5.64	8.28	8.11



Şekil 3. Veri Setlerinde Çıkarılan Niteliklerin Sayısı (%) ve Sınıflandırma Doğruluk Değerleri (The Reduced Number of Attributes in Datasets and the Success Rate of the Results)

Şekil 3'te her veri seti için nitelik çıkarma/boyut azaltma yapılmadan önce k-nn ve AGDE-knn uygulanarak elde edilen ortalama sınıflandırma doğruluk değerleri, o veri setinin ortalama niteliklerinin yüzde kaçının çıkarıldığı ve bu çıkarılan niteliklerden sonra k-nn ve AGDE-knn algoritmaları uygulanarak elde edilen ortalama sınıflandırma doğruluk değerleri verilmiştir.

Şekil 3-a'ya bakılarak veri setinin niteliklerinin %96'sı çıkarıldığında performansının düşmediği hatta k-nn algoritmasında iyileşme olduğu görülmüştür. Şekil 3-b'ye bakılarak veri setinin niteliklerinin %51'i çıkarıldığında algoritmaların performansının kötüleşmediği hatta iki algoritma için sınıflandırma doğruluk değerlerinin arttığı görülmüştür. Şekil 3-c'ye bakıldığında veri setinde niteliklerin %81'i çıkarıldığında sınıflandırma performansının kötüleşmediği hatta AGDE-knn algoritmasında iyileşme olduğu görülmektedir. Şekil 3-d'ye bakıldığında veri setinde niteliklerin %71'i çıkarıldığında doğruluk değerlerinin düşmediği hatta her iki algoritmanın performansında iyileşme olduğu görülmüştür.

4. Sonuç ve Tartışma (Result and Discussion)

Bu çalışmada, AGDE-knn ile sezgisel olarak problem boyutunun azaltılması (nitelik seçimi) ve bu boyut azaltma sonucunda knn ve AGDE-knn 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 dört farklı veri seti üzerinde AGDE-knn 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.

Teşekkür (Acknowledgement)

Bu çalışmada yürütülen faaliyetler, 2020 yılında TÜBİTAK 2209-A Üniversite Öğrencileri Yurt İçi Araştırma Projeleri Destek Programı kapsamında 1919B011904088 numaralı proje olarak TÜBİTAK tarafından desteklenmiştir.

Çıkar Çatışması (Conflict of Interest)

Yazarlar tarafından herhangi bir çıkar çatışması beyan edilmemiştir. No conflict of interest was declared by the authors.

Kaynaklar (References)

- Abualigah, L. M., Khader, A. T., Al-Betar, M. A., & Alomari, O. A. (2017). Text feature selection with a robust weight scheme and dynamic dimension reduction to text document clustering. *Expert Systems with Applications*, 84, 24-36.
- Aljarah, I., Mafarja, M., Heidari, A. A., Faris, H., & Mirjalili, S. (2020). Multi-verse optimizer: theory, literature review, and application in data clustering. In *Nature-Inspired Optimizers* (pp. 123-141). Springer, Cham.
- Balint Antal, Andras Hajdu: An ensemble-based system for automatic screening of diabetic retinopathy, *Knowledge-Based Systems* 60 (April 2014), 20-27.
- Booker, L. B., Goldberg, D. E., & Holland, J. H. (1989). Classifier systems and genetic algorithms. *Artificial intelligence*, 40(1-3), 235-282.
- Cerrada, M., Aguilar, J., Altamiranda, J., & Sánchez, R. V. (2019). A hybrid heuristic algorithm for evolving models in simultaneous scenarios of classification and clustering. *Knowledge and Information Systems*, 61(2), 755-798.
- Dash, M., & Liu, H. (1997). Feature selection for classification. *Intelligent data analysis*, 1(3), 131-156.
- Fausto, F., Reyna-Orta, A., Cuevas, E., Andrade, Á. G., & Perez-Cisneros, M. (2020). From ants to whales: metaheuristics for all tastes. *Artificial Intelligence Review*, 53(1), 753-810.
- Inbarani, H. H., Bagyamathi, M., & Azar, A. T. (2015). A novel hybrid feature selection method based on rough set and improved harmony search. *Neural Computing and Applications*, 26(8), 1859-1880.
- Johnson, B., 2013. High resolution urban land cover classification using a competitive multi-scale object-based approach. *Remote Sensing Letters*, 4 (2), 131-140.
- Kahraman, H. T. (2016). A novel and powerful hybrid classifier method: Development and testing of heuristic k-nn algorithm with fuzzy distance metric. *Data & Knowledge Engineering*, 103, 44-59.
- Kahraman, H. T., Bayindir, R., & Sagioglu, S. (2012). A new approach to predict the excitation current and parameter weightings of synchronous machines based on genetic algorithm-based k-NN estimator. *Energy Conversion and Management*, 64, 129-138.
- Kahraman, H. T., Sagioglu, S., & Colak, I. (2013). The development of intuitive knowledge classifier and the modeling of domain dependent data. *Knowledge-Based Systems*, 37, 283-295.
- Kulkarni, A. J., Singh, P. K., Satapathy, S. C., Kashan, A. H., & Tai, K. (Eds.). (2019). *Socio-cultural Inspired Metaheuristics* (Vol. 828). Springer.
- Kwak, N., & Choi, C. H. (2002). Input feature selection for classification problems. *IEEE transactions on neural networks*, 13(1), 143-159.
- Li, K., Cao, X., Ge, X., Wang, F., Lu, X., Shi, M., ... & Chang, S. (2020). Meta-Heuristic Optimization Based Two-stage Residential Load Pattern Clustering Approach Considering Intracluster Compactness and Inter-cluster Separation. *IEEE Transactions on Industry Applications*.
- Lin, K. C., Zhang, K. Y., Huang, Y. H., Hung, J. C., & Yen, N. (2016). Feature selection based on an improved cat swarm optimization algorithm for big data classification. *The Journal of Supercomputing*, 72(8), 3210-3221.
- Mohamed, A. W., & Mohamed, A. K. (2019). Adaptive guided differential evolution algorithm with novel mutation for numerical optimization. *International Journal of Machine Learning and Cybernetics*, 10(2), 253-277.
- O.L. Mangasarian, W.N. Street and W.H. Wolberg. Breast cancer diagnosis and prognosis via linear programming. *Operations Research*, 43(4), pages 570-577, July-August 1995.
- Özkaraca, O., & Keçebaş, A. (2019). Performance analysis and optimization for maximum exergy efficiency of a geothermal power plant using gravitational search algorithm. *Energy Conversion and Management*, 185, 155-168.
- Özkaraca, O., Keçebaş, A., & Demircan, C. (2018). Comparative thermodynamic evaluation of a geothermal power plant by using the advanced exergy and artificial bee colony methods. *Energy*, 156, 169-180.
- Özkaraca, O., Keçebaş, P., Demircan, C., & Keçebaş, A. (2017). Thermodynamic optimization of a geothermal-based organic Rankine cycle system using an artificial bee colony algorithm. *Energies*, 10(11), 1691.
- R. Alizadehsani, J. Habibi, M. J. Hosseini, H. Mashayekhi, R. Boghrati, A. Ghandeharioun, et al., 'A data mining approach for diagnosis of coronary artery disease,' *Computer Methods and Programs in Biomedicine*, vol. 111, pp. 52-61, 2013/07/01/2013.
- Ramos-Figueroa, O., Quiroz-Castellanos, M., Mezura-Montes, E., & Schütze, O. (2020). Metaheuristics to solve grouping problems: A review and a case study. *Swarm and Evolutionary Computation*, 100643.
- Rouhi, A., & Nezamabadi-pour, H. (2017, March). A hybrid feature selection approach based on ensemble method for high-dimensional data. In *2017 2nd Conference on Swarm Intelligence and Evolutionary Computation (CSIEC)* (pp. 16-20). IEEE.
- Santhanam, T., & Padmavathi, M. S. (2015). Application of K-means and genetic algorithms for dimension reduction by integrating SVM for diabetes diagnosis. *Procedia Computer Science*, 47, 76-83.

- Sibalija, T. V. (2019). Particle swarm optimisation in designing parameters of manufacturing processes: A review (2008–2018). *Applied Soft Computing*, 84, 105743.
- Storn, R., & Price, K. (1997). Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4), 341-359.
- Tang, J., Alelyani, S., & Liu, H. (2014). Feature selection for classification: A review. *Data classification: Algorithms and applications*, 37.
- Thangavel, K., & Pethalakshmi, A. (2009). Dimensionality reduction based on rough set theory: A review. *Applied Soft Computing*, 9(1), 1-12.
- Tran, B., Xue, B., & Zhang, M. (2018). Variable-length particle swarm optimization for feature selection on high-dimensional classification. *IEEE Transactions on Evolutionary Computation*, 23(3), 473-487.
- Xue, B., Zhang, M., & Browne, W. N. (2012). Particle swarm optimization for feature selection in classification: A multi-objective approach. *IEEE transactions on cybernetics*, 43(6), 1656-1671.
- Yadav, M., & Prakash, V. P. (2020). A Comparison of the Effectiveness of Two Novel Clustering-Based Heuristics for the p-Centre Problem. In *Advances in Data and Information Sciences* (pp. 247-255). Springer, Singapore.
- Yilmaz, C., Kahraman, H. T., & Söyler, S. (2018). Passive mine detection and classification method based on hybrid model. *IEEE Access*, 6, 47870-47888.
- Yusta, S. C. (2009). Different metaheuristic strategies to solve the feature selection problem. *Pattern Recognition Letters*, 30(5), 525-534.
- Zhou, Q., Benlic, U., Wu, Q., & Hao, J. K. (2019). Heuristic search to the capacitated clustering problem. *European Journal of Operational Research*, 273(2), 464-487.