

**TÜBİTAK–2209-A ÜNİVERSİTE ÖĞRENCİLERİ ARAŞTIRMA PROJELERİ DESTEĞİ PROGRAMI**

**ARAŞTIRMA ÖNERİSİ FORMU**

2019

Ekim (Güz) Dönem Başvurusu

**A. GENEL BİLGİLER**

|  |
| --- |
| **Başvuru Sahibinin Adı Soyadı:** Büşra Aras, Orhun Yıldız |
| **Araştırma Önerisinin Başlığı:** Sınıflandırma Problemleri İçin Meta-Sezgisel Boyut Azaltma Aracının Tasarımı Ve Uygulaması |
| **Danışmanın Adı Soyadı:** Doç. Dr. Hamdi Tolga Kahraman |
| **Araştırmanın Yürütüleceği Kurum/Kuruluş:** Karadeniz Teknik Üniversitesi |

**ÖZET**

Veri madenciliği konusunun son dönemlerde büyük veri olarak adlandırıldığı ve evrimleştiği görülmektedir. Bu durum teknolojideki gelişmenin, veri üreten cihazların sayısındaki artışın ve veri işleme mekanizmalarının/algoritmaların çeşitlenmesinin bir neticesidir. Her geçen gün geçmişe nazaran çok daha fazla verinin, çok daha hızlı bir şekilde toplanabildiği, kaydedilebildiği ve işlenebildiği bir dönem yaşanmaktadır. Verinin toplanması ve işlenmesinde gelişmeler yaşanmış olsa da her geçen gün veri sayısındaki artışın giderek büyüyen bir veriye yol açtığı ve tüm gelişmelere rağmen veri işlemenin halen daha zor bir iş olduğu açıktır. Veri işleme sürecinde iyileşmeler olmuş olsa da büyüyen veri karşısında makinelerin işlem yetenekleri halen daha ciddi şekilde yetersiz kalabilmektedir. Tüm bunlar büyük veri işleme ve veri madenciliği konularında yeni çözümlere ve çalışmalara ihtiyaç duyulduğunu göstermektedir. Bu proje önerisinin amacı büyük verinin işlenmesinde karşılaşılan önemli bir problem olan boyut azaltma üzerine etkili ve modern bir yöntem geliştirmektir. Veri madenciliğinin en önemli problem türlerinden olan sınıflandırma problemlerinin etkili ve hızlı bir şekilde çözümlenebilmesini sağlamak amacıyla sezgisel boyut azaltma algoritmasının geliştirilmesi planlanmaktadır.

**Anahtar Kelimeler:** Yapay zekâ, veri madenciliği, boyut azaltma, nitelik seçimi, meta-sezgisel optimizasyon

1. **ÖZGÜN DEĞER** 
   1. **Konunun Önemi, Araştırma Önerisinin Özgün Değeri ve Araştırma Sorusu/Hipotezi**

*Konunun önemi:* Veri madenciliği uygulamalarında karşılaşılan önemli problemlerden biri problem boyutunun büyümesi ile test orantılı olarak veri işleme performansının azalmasıdır. Problem boyutunun artması halinde algoritmaların işlem yükü de artmaktadır. Bu durum tüm algoritmalar için geçerlidir. Algoritmaların karar verme sürecini olumsuz etkileyen bu problemin etkili bir şekilde çözümlenebilmesi için çok sayıda çalışma yapılmıştır [1-19]. Bu çalışmaların başlıca amacı, veri madenciliği uygulamalarının karar verme doğruluğunu/hassasiyetini kötüleştirmeden cevaplama hızını iyileştirmektir. Özellikle 7/24 çevrimiçi (on-line) çalışan canlı sistemlerde veri madenciliği uygulamalarının karar verme hızlarının veri akış hızına uygun olması karşılanması zor bir gereksinimdir. Bu durum Endüstri 4.0 gibi canlı veri akışının kesintisiz sürdüğü uygulamalarda elzemdir. Haliyle problem boyutunun azaltılması yoluyla algoritmaların işlem yükünün azaltılması, kararların daha hızlı bir şekilde verilmesi açısından önemli ve etkili bir çözüm yolu olarak görülmektedir. Boyut indirgeme konusundaki referans çalışmaların son yıllarda artması ve bu çalışmaların nitelikli dergilerde yayınlanması konunun önemini göstermektedir. Bu proje önerisi, veri madenciliği açısından böylesi önemli bir problemin çözümünü konu almaktadır.

*Araştırma önerisinin özgün değeri:* bu proje önerisi yapay zekânın önemli bir problemi olan boyut indirgeme üzerinedir. Boyut indirgeme bilinen bir konu olsa da çözülmesi zor ve dinamik bir problemdir. Öyle olduğu içindir ki uzunca yıllardır üzerinde çalışılmakta ve etkili ve genel bir çözüm yolunun geliştirilmesi üzerine araştırmalar devam etmektedir. Araştırmacılar temel bileşen analizini kullanarak [10-14] bu konu üzerinde epeyce çalışmalar yürütmüşlerdir. Günümüzde ise sezgisel arama algoritmalarının boyut indirgeme problemine tatbik edilmeye çalışıldığı ve bu konudaki araştırmaların önemli miktarda arttığı görülmektedir [15-19]. Ancak literatürdeki çalışmaların önemli bir eksikliği boyut azaltma çalışmalarının problem bazlı olarak yürütülmesidir. Bu proje önerisinde ise literatürden farklı olarak boyut azaltma problemine uygulama alanından bağımsız bir şekilde çözüm geliştirmektir. Uygulama alanından bağımsız bir yaklaşım geliştirmek için yazılım tasarım prensiplerinden ve problem bağımlılığı en az olan yapay zeka algoritmalarından faydalanılacaktır. Öncelikle literatürdeki çalışmaların aksine, meta-sezgisel arama biriminde, kullanıcı tanımlı parametreye ihtiyaç duymayan modern ve güçlü bir algoritma olan Ortak Yaşayan Organizmalar (Symbiotic Organism Search, SOS) [20] tatbik edilecektir. Boyut azaltmanın etkisini hızlı bir şekilde ortaya çıkarabilmek için sınıflandırma tekniği olarak yapay sinir ağları tatbik edilecektir. Boyut azaltmanın etkisini modelleyebilmek için ise k-en yakın komşu algoritmasından faydalanılacaktır. Tüm bu özellikler, gerçekleştirilmesi planlanan projeye özgün çözümler olarak literatüre kazandırılacaktır.

*Araştırma sorusu/Hipotezi:*Problem boyutu nasıl azaltılabilir? Problem boyutunu azaltmada kullanılacak olan yöntemler nelerdir? Problem boyutunu azaltma işleminde yapay zekâ tabanlı algoritmalar kullanılabilir mi? Problem boyutu azaltıldığı takdirde performans düşüşü olur mu, eğer olursa kabul edilebilir oranda mı olur? Öznitelik seçimi için meta sezgisel arama algoritmaları kullanılabilir mi? SOS algoritmasının kullanıcı tanımlı tasarım parametresi gerektirmemesi genel ve etkili bir yöntem geliştirmede faydalı olabilir mi? SOS algoritmasının başarısı boyut azaltma sürecinde de etkisini gösterir mi? Geliştirilecek olan bu yöntem, var olan diğer yöntemlerden (örn: Temel Bileşen Analizi) daha etkili ve hızlı bir biçimde boyut azaltma işlemini gerçekleştirebilir ve performans anlamında artış sağlayabilir. Bu hipotezin deneysel çalışmalar yoluyla araştırılmasına ve doğrulanmasına ihtiyaç vardır.

* 1. **Amaç ve Hedefler**

Araştırma önerisinin amacı ve hedefleri açık, ölçülebilir, gerçekçi ve araştırma süresince ulaşılabilir nitelikte olacak şekilde yazılır.

Önerilen projede amaç, çok boyutlu sınıflandırma problemleri için uygulama alanına bağlı kalmaksızın boyut indirgeme işlemini etkili bir şekilde yerine getiren meta-sezgisel optimizasyon tabanlı bir boyut indirgeme aracı geliştirmektir. Bu amaca yönelik olarak hazırlanmış açık ve ölçülebilir hedefler aşağıda sıralanmaktadır.

* Geliştirilecek uygulamanın kapsamı: Problem tipi sınıflandırma ve problemin bağımsız değişkenlerinin veri türleri sayısal olacaktır. Problem boyutu alt ve üst sınırları [5, 500] olacaktır.
* Web üzerinden erişilebilir bir uygulama geliştirilecektir. Kapsamda belirtilen özelliklere sahip veri setleri için kullanıcının algoritma bilgisine ihtiyaç duyulmadan en uygun problem boyutu belirlenecektir.
* Test ve doğrulama çalışmalarının sonuçları literatürde standart hale gelmiş karşılaştırma matrisleri (confusion matrix) [22] ile sunulacaktır.
* Boyut azaltma işlemi sonucunda sistem tarafından eğitilen bir yapay sinir ağı modeli yaratılacak ve kullanıcının bu modeli kaydetmesine imkân tanınacaktır.
* Boyut azaltma işlemi neticesinde problem için optimum tasarım değişkenleri belirlenerek kullanıcılara sunulacaktır.

1. **YÖNTEM**

Projede kullanılacak algoritmalar, güncel bir meta-sezgisel arama algoritması olan Ortak Yaşayan Organizmalar Algoritması (Symbiotic Organism Search, SOS) [20], k-en yakın komşu algoritması (k-nearest neighbour, k-nn) [21-23], yapay sinir ağları [24-25] ve sezgisel k-nn sınıflandırma algoritmalarıdır [22-23, 26]. Buna göre aşağıdaki alt bölümlerde öncelikle veri madenciliği konusu kısaca ele alınmakta ve sırasıyla meta-sezgisel arama süreci, k-nn algoritması, yapay sinir ağları ve sezgisel k-nn algoritmaları hakkında bilgi verilmektedir.

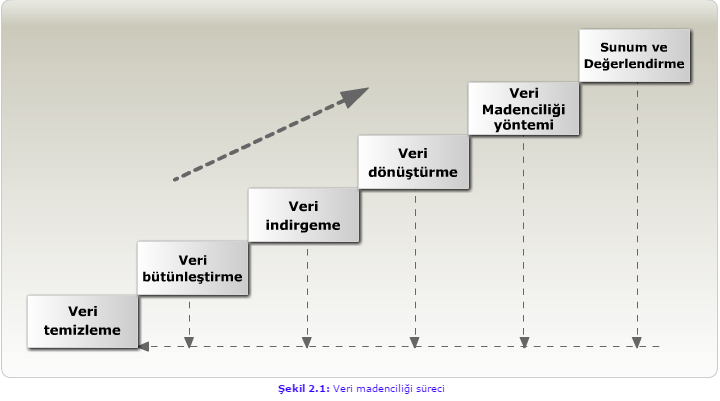
* 1. **Veri Madenciliği**

“Teknolojideki baş döndürücü gelişmeler birçok yeni durumla karşılaşmamıza neden olmuştur. Veri depolayan cihaz sayısı artmıştır. Dolayısıyla veri sayısında da olağanüstü bir artış meydana gelmiştir. Bu durum istatiksel yöntemlere dayalı geleneksel veri madenciliği yöntemlerinin etkisiz kalmasına yol açmıştır. Bunun sonucunda yapay zekâ tabanlı veri madenciliği yöntemlerine ve büyük veri işleme yeteneklerine ihtiyaç duyulmuştur [21].

Örneğin bir mağazada satışlar ve müşterilerle ilgili her türlü veri sayısal ortamda yerini almaktadır. Üstelik günlük tüm veriler sayısal ortamda saklanmaktadır. Binlerce müşterisi olan bir mağaza her gün çok sayıda veri üretmek zorunda kalmaktadır. Böylece ilgili firma bilgisayarlarında çok büyük miktarda veri birikmektedir. Bir kredi kartı firmasında yada telefon iletişimi yapan bir firmada günlük bazda biriken verilerin miktarı çok büyük olacaktır. Bilişim teknolojisi bu devasa verileri saklamaya yeterli olabilir. Ancak bu veriler ne işe yarayacaktır? Bu verilerden firma bazı avantajlar kazanabilecek midir? Biriken veri gerçek anlamda “bilgiye” dönüştürülebilecek midir? Bu tür sorulara olumlu yanıt vermek mümkündür. Büyük ölçekli verilerden yararlanarak firma karar destek sistemleri oluşturulabilir. Bu veriler üzerinde çözümlemeler yapılarak özellikle stratejik seviyedeki kararlara destek sağlanabilir. Veriler üzerinde çözümlemeler yapmak amacıyla çeşitli istatistiksel ve matematiksel yöntemler kullanılabilir. Ancak veri sayısı arttıkça sorunlar ortaya çıkacaktır. Özellikle ilişkisel veri tabanları üzerinde bu çözümlemeleri yapmak zorlaşacaktır. Bu tür veriler üzerinde çözümlemeleri yapabilmek için hem yeni veri tabanı kavramlarına hem de yeni çözümleme yöntemlerine gereksinim duyulmaktadır. Veriyi yönetmek için “Veri Ambarı” ve verileri çözümleyerek “yararlı bilgiye” erişilmesini sağlayan “Veri Madenciliği” kavramları ortaya atılmıştır.

“Veri madenciliği; önceden bilinmeyen, gizli, anlamlı ve yararlı örüntülerin büyük ölçekli veri tabanlarından otomatik biçimde elde edilmesini sağlayan veri tabanlarındaki özbilgi keşif ve analiz süreci içerisinde yer alan bir adımdır. Veri tabanlarındaki özbilgi keşif ve analiz sürecinde yer alan adımlar Şekil 1’de görülmektedir”. Veri madenciliğini bir süreç olarak değerlendirmek gerekiyor. Söz konusu süreç aşağıda belirtilen adımları içermektedir [27-33]:

1. Veri temizleme: Bazı uygulamalarda, üzerinde çözümleme yapılacak verilerin istenen özelliklere sahip olmadığı görülebilir. Örneğin eksik verilerle ve uygun olmayan verilerin oluşturduğu tutarsız verilerle karşılaşılabilir. Veri tabanında yer alan tutarsız ve hatalı verilere gürültü olarak değerlendirilmektedir. Bu gibi durumlarda verinin söz konusu sorunlardan temizlenmesi gerekecektir. Eksik verilerin yerine yenileri belirlenerek konulmalıdır. Bunun için aşağıda belirtilen yöntemlerden biri kullanılabilir. Eksik değer içeren kayıtlar veri kümesinden atılabilir. Kayıp değerlerin yerine bir genel sabit kullanılabilir. Bütün kayıp değerler için aynı sabit kullanılabilir. Örneğin "bilinmiyor" değeri bu eksik veri yerine kullanılabilir. Ancak bütün değişkenlerde kayıp değerler yerine aynı sabit değerin kullanımı sorun yaratacaktır. Değişkenin tüm verileri kullanılarak ortalaması hesaplanır ve eksik değer yerine bu değer kullanılabilir. Değişkenin tüm verileri yerine, sadece bir sınıfa ait örneklerin değişken ortalaması hesaplanarak eksik değer yerine kullanılabilir. Verilere uygun bir tahmin yapılarak, örneğin regresyon ya da karar ağacı modeli kurularak eksik değer tahmin edilebilir ve eksik değer yerine kullanılabilir.
2. Veri bütünleştirme: Farklı veri tabanlarından ya da veri kaynaklarından elde edilen verilerin birlikte değerlendirmeye alınabilmesi için farklı türdeki verilerin tek türe dönüştürülmesi yani bütünleştirilmesi söz konusu olacaktır. Eğer veri madenciliği uygulaması için bir veri ambarı altyapısı hazırlanmış ise söz konusu veri bütünleştirme işleminin yapılmış olması gerekmektedir. Ancak böyle bir yapı yoksa söz konusu veri bütünleştirme işleminin doğrudan veri madenciliğine esas oluşturacak veriler üzerine uygulanması gerekecektir.
3. Veri indirgeme: Veriyi indirgeme aşamasında verileri, çok boyutlu veri küpleri biçimine dönüştürmek söz konusu olabilir. Böylece çözümlemeler sadece belirlenen boyutlara göre yapılır. Veriler arasında bir seçme işlemi yapılarak, gereksiz veriler veri tabanından çıkarılır ve boyut azaltılması sağlanabilir. Veri sıkıştırma aşamasında, büyük veri kümelerinin sıkıştırılarak daha az yer işgal etmeleri sağlanır. Örnekleme aşamasında ise, büyük veri topluluğu yerine onu temsil eden daha küçük veri kümelerinin oluşturulması amaçlanır. Genelleme verilerin tek tek değil, genel kavramlarla ifade edilmesini sağlar.
4. Veri dönüştürme: Veriyi bazı durumlarda veri madenciliği çözümlemelerine aynen katmak uygun olmayabilir. Değişkenlerin ortalama ve varyansları birbirlerinden önemli ölçüde farklı olduğu takdirde büyük ortalama ve varyansa sahip değişkenlerin diğerleri üzerindeki baskısı daha fazla olur ve onların rollerini önemli ölçüde azaltır. Ayıca değişkenlerin sahip olduğu çok büyük ve çok küçük değerler de çözümlemelerin sağlıklı biçimde yapılmasını engeller. Bu nedenle bir dönüşüm yöntemi uygulayarak söz konusu değişkenlerin normalleştirilmesi veya standartlaştırılması uygun bir yol olacaktır.



Şekil 1. Veri madenciliği sürecinin adımları

1. Veri madenciliği algoritmasını uygulama: bu aşamada veriden bilgiye dönüşüm için kullanılacak olan algoritmanın tasarımı gerçekleştirilir. Bağımlı ve bağımsız değişkenler arasındaki analitik ilişki bu yolla modellenmiş olur. Yapay sinir ağları, k-nn sınıflandırıcı, sezgisel arama algoritmaları, karar ağaçları, derin sinir ağları, yalın bayes, bulanık mantık bu algoritmalardan bir kaçıdır [22-28, 34].
2. Sonuçları sunum ve değerlendirme: veri madenciliğinde verinin bilgiye dönüştürülmesi önemli olduğu kadar bu dönüşüm neticesinin de kullanıcılar tarafından kolay ve etkili bir şekilde anlaşılabilmesi de o kadar önemlidir. Bu amaçla sınıflandırma problemlerinde özellikle karşılaştırma matrisleri (confusion matrix) kullanılmaktadır [3].
   1. **Meta-sezgisel arama süreci**

Yapay zekânın bir türü olan Meta-sezgisel Arama (MSA) algoritmalarını geliştirme çalışmaları 1950’li yıllara dayanmaktadır. Michigan üniversitesinde Prof. John Holland ve öğrencilerinin geliştirdikleri Genetik Algoritma bu çalışmalara hız kazandırmıştır [35]. Son yıllarda gerek algoritma geliştirme çalışmalarında gerekse de problemlere tatbik edilmeleri hususunda elde edilen başarılar meta-sezgisel algoritmaların önemini giderek artırmaktadır [36-40]. MSA algoritmaları yapay zekanın uygulandığı her alanda, tahmin, kümeleme, sınıflandırma gibi problemleri çözümlemek için melez algoritmaların geliştirilmesinde ve asıl olarak optimizasyon problemlerinin çözümünde yaygın bir şekilde kullanılmaktadırlar. Maliyetleri azaltmanın ve verimliliği artırmanın kritik önem kazandığı çağımızda süreçleri ve sistemleri optimum şekilde modellemenin etkili yollarından biri olarak meta-sezgisel algoritmalara başvurulmaktadır. Enerji, inşaat, pazarlama, üretim, bilgi teknolojileri, havacılık ve uzay sanayii gibi birçok alanda binlerce sistem ve sürecin optimizasyonunda ve Endüstri 4.0 gibi modern otomasyon sistemlerinin ve uygulamalarının geliştirilmesinde meta-sezgisel optimizasyon tekniklerinden faydalanılmaktadır. Büyük ve karmaşık bir problem uzayında arama yapmanın etkili bir yolu MSA algoritmalarını kullanmaktır. MSA algoritmalarının doğadan kaynaklı bileşenlerinin yetenekleri ve özellikleri farklı olsa da bir meta-sezgisel arama süreci temel olarak aynı adımlardan oluşur. Bu adımlar Algoritma 1’de verilmektedir.

**Algoritma 1.** Meta-sezgisel arama sürecinin temel adımları

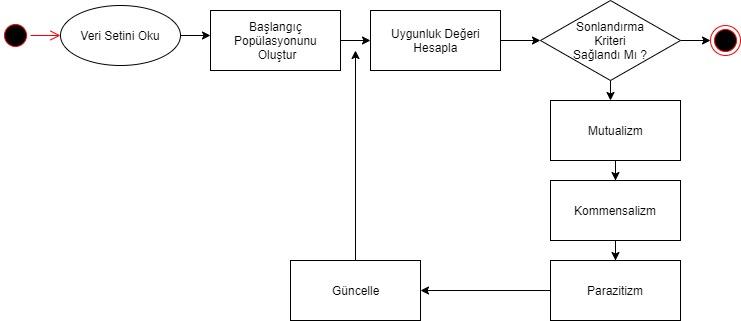
|  |
| --- |
| i) Problemin yaratılması (uygunluk fonksiyonunun, ceza fonksiyonunun tanımlanması)  ii) Çözüm adayının tasarımı ve çözüm adayları topluluğunun yaratılması  iii) Adayların uygunluk değerlerinin hesaplanması  iv) İteratif süreç (buluşsal arama)   * + Komşuluk Araması   + Çeşitliliğin Sağlanması   + Çözüm adayı setinin güncellenmesi   v) Sonlandırma kriteri sağlandı mı?   * + Hayır (Adım iv’e dön)   + Evet (arama sürecini sonlandır ve en iyi çözüm adayını kaydet) |

Algoritma 1’de verilen (i, ii, iii ve v) numaralı adımlar bütün MSA algoritmaları için aynıdır. (iv) numaralı adım ise tüm MSA için farklıdır. (iv) numaralı adımda arama algoritmasına özgü operatörler/işlemler uygulanmaktadır. Arama sürecinin başarısı bu operatörlerin yeteneklerine bağlıdır. (iv) numaralı adımda verilen çözüm adayları topluluğunun güncellenmesinde temel olarak iki farklı yöntem kullanılmaktadır. Bunlar çözüm adaylarının arama uzayına normal/gauss dağılımı ile yerleştirilmesi ve rastgele yerleştirilmesidir. MSA algoritmaları ile ilgili çalışmalar incelendiğinde bu dağılım tiplerinden biriyle algoritmaların geliştirildiği ve test edildiği görülmektedir.

* 1. **Ortak yaşayan organizmalar algoritması (Symbiotic Organism Search, SOS)**

Meta-sezgisel optimizasyon algoritmalarından biri olan SOS, diğer algoritmalar arasında basit ve güçlü yapısı ile dikkat çeken bir algoritmadır. SOS algoritması problem uzayındaki genel en iyi çözümü bulmak için popülasyon-tabanlı bir iteratif yöntemdir. SOS algoritmasında popülasyona verilen isim ekosistemdir. Popülasyonun ilk oluşturulması aşamasında arama uzayında bir grup organizma rastgele olarak başlatılır. Her bir organizma problem için bir aday çözümü temsil etmektedir. SOS algoritmasında, ekosistemdeki iki organizma arasındaki biyolojik etkileşimi modelleyerek yeni çözümler üretilir. Gerçek yaşamdaki biyolojik etkileşimleri model olarak kullanan algoritma üç adımdan oluşur. Bu etkileşimler sırasıyla mutualizm, kommensalizm ve parazitizmdir.

Mutualizm iki ayrı türün karşılıklı yarar sağladıkları simbiyotik ilişkiyi canlandırır. Kommensalizm iki organizmadan birinin yarar gördüğü diğerinin ise ne yarar ne de zarar gördüğü simbiyotik ilişkidir. Parazitizm ise iki organizmadan birinin yarar görürken diğerinin zarar gördüğü simbiyotik ilişkidir [20].

****

Şekil 2. SOS algoritması akış diyagramı

* 1. **k-nn sınıflandırma algoritması**

k-nn algoritması uzaklığa dayalı ve sınıflandırma algoritmasıdır. Sınıflandırma, önceden elde ettiğimiz bilgilerin veya verilerin hangi sınıftan olduğu biliniyorsa, yeni gelen verinin hangi sınıfa ait olacağının belirlenmesi işlemidir. Uzaklığa dayalı algoritmalardan en bilineni ve en yaygın kullanılanıdır. Sınıflandırma yapılırken eldeki verilerin birbirlerine olan uzaklığı veya benzerliği kullanılır. Veriler arasındaki mesafe ölçülülerken en çok kullanılan yöntemler Öklit, Manhattan ve Minkovski uzaklık metrikleridir. k-nn algoritmasında öncelikle gözlemler arasındaki uzaklıklar belirlenir. Gözlemler arasındaki uzaklıkların belirlenmesinde herhangi bir uzaklık bağıntısı kullanılabilir. Sınıflandırma yapılırken veri tabanındaki her bir kaydın diğer kayıtlara olan uzaklığı hesaplanır [21-23].

1. **Öklit Uzaklığı**

Uygulamada en çok kullanılan uzaklık bağıntısı Öklid uzaklık bağıntısıdır. p değişken sayısı olmak üzere, i,j=1,2,...,n ve k=1,2,...,p için Öklid uzaklığı şu şekilde olacaktır:



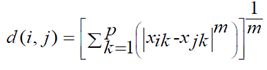
1. **Manhattan Uzaklığı**

Diğer bir uzaklık bağıntısı ise Manhattan uzaklığıdır. Bu uzaklık, gözlemler arasındaki mutlak uzaklıkların toplamı alınarak hesaplanır. Söz konusu uzaklık şu şekilde ifade edilir:



1. **Minkovski Uzaklığı**

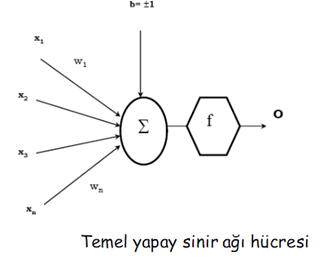
Gözlemler arasındaki uzaklıkların hesaplanmasında Minkovski uzaklık bağıntısı da kullanılabilmektedir:



Uygulamada “K” değeri önceden seçilir. Bu değerin büyük seçilmesi birbirine benzemeyen noktaların bir araya toplanmasına, çok küçük seçilmesiyse birbirine benzediği, yani aynı sınıfın noktaları oldukları halde, bazı noktaların ayrı sınıflara konmasına ya da o tür noktalar için ayrı sınıfların açılmasına neden olur. K-NN algoritması kolay uygulanabilen bir algoritma olmasının yanı sıra hesaplama maliyeti gerçekten çok yüksektir çünkü her bir sorgu örneğinin tüm eğitim örneklerine olan uzaklığını hesaplamak gerekmektedir. Bazı indeksleme metotları ile bu maliyet azaltılabilir.

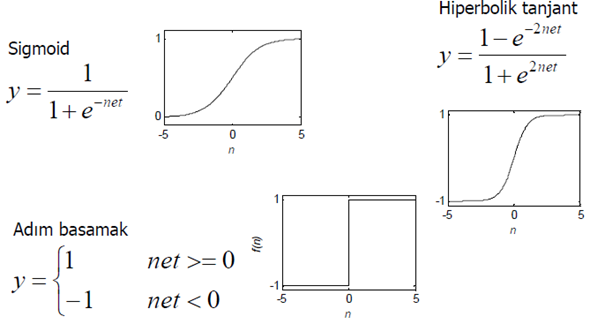
* 1. **Yapay sinir ağları**

Yapay sinir ağları (YSA), beynin bir işlevi yerine getirme yöntemini modellemek için tasarlanan bir sistem olarak tanımlanabilir. YSA, yapay sinir hücrelerinin birbirleri ile çeşitli şekillerde bağlanmasından oluşur ve genellikle katmanlar şeklinde düzenlenir. Beynin öğrenme sürecine uygun olarak YSA, bir öğrenme sürecinden sonra bilgiyi toplama, hücreler arasındaki bağlantı ağırlıkları ile bu bilgiyi saklama ve genelleme yeteneğine sahip paralel dağıtılmış bir işlemcidir. Öğrenme süreci, arzu edilen amaca ulaşmak için YSA ağırlıklarının yenilenmesini sağlayan, öğrenme algoritmalarını içerir. Temel bir yapay sinir ağı hücresinde; girişler, ağırlıklar, toplama fonksiyonu, aktivasyon fonksiyonu, çıkışlar bulunur.

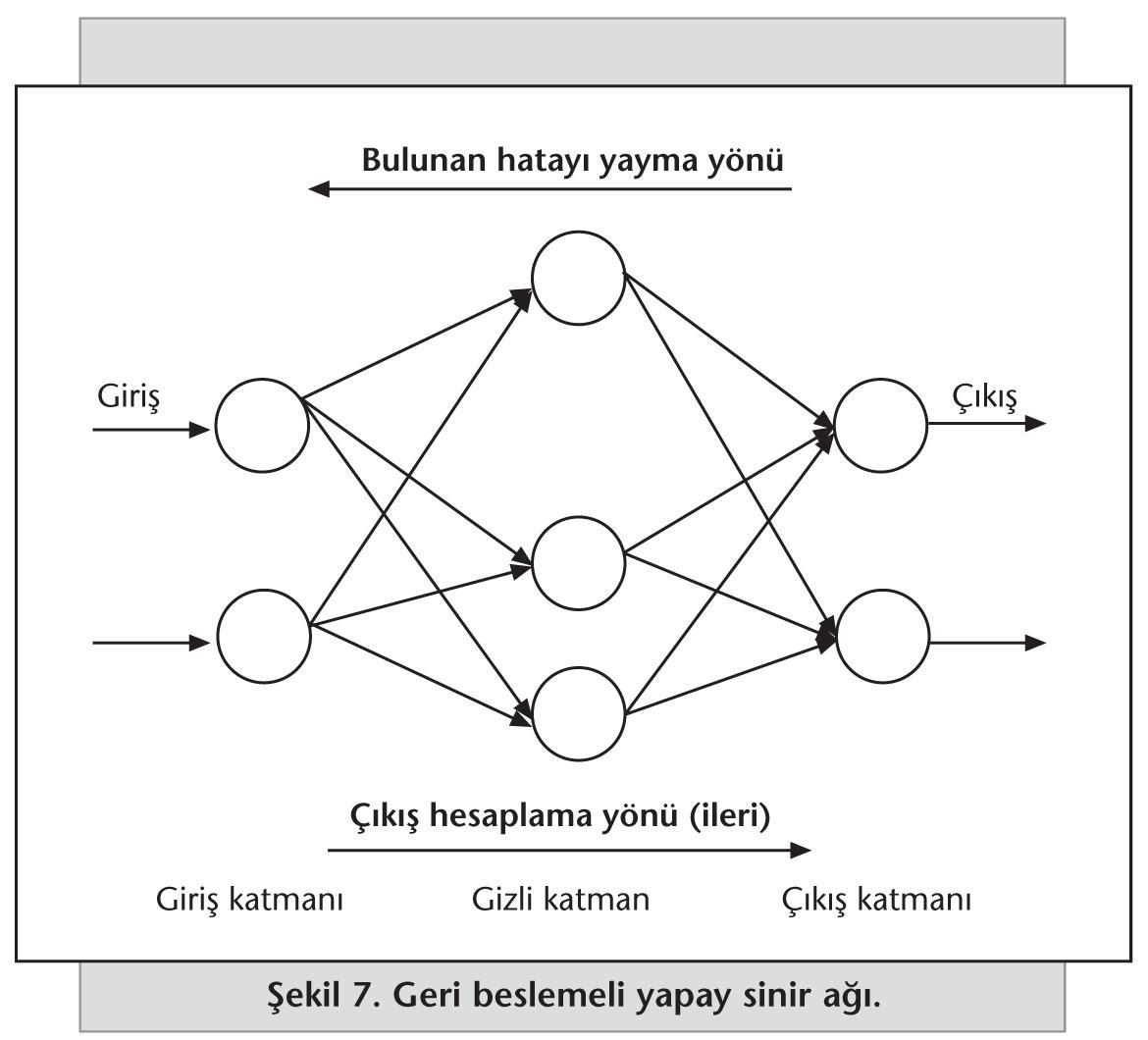


Dış ortamdan veya diğer hücrelerden alınan girdiler, ağırlıklar yardımıyla hücreye bağlanır. Toplam fonksiyonu ile net girdi hesaplanır. Net girdinin aktivasyon fonksiyonundan geçirilmesiyle net çıktı hesaplanır. Bu işlem aynı zamanda bir hücrenin çıkışını verir. Her bağlantının bir ağırlık değeri vardır. YSA, kendine örnekler geliştirildikçe bu ağırlık değerlerini değiştirir. Hedef, ağa gösterilen örnekleri için doğru çıkışları verecek ağırlıkları bulmaktır. En yaygın toplama fonksiyonu ağırlıklar toplamıdır.

Bir yapay sinir ağında, her hücre elemanı bağımsız olarak farklı bir toplama fonksiyonuna sahip olabileceği gibi hepsi aynı toplama fonksiyonuna da sahip olabilir. Aktivasyon fonksiyonu, hücreye gelen net girdiyi işleyerek hücrenin bu girdiye karşılık üreteceği çıktıyı belirler. Genelde aktivasyon fonksiyonu doğrusal olmayan bir fonksiyondur ve toplama fonksiyonunda olduğu gibi ağın hücre elemanlarının hepsi aynı fonksiyonu kullanması gerekmez. En çok kullanılan aktivasyon fonksiyonu sigmoid fonksiyondur.



Yapay sinir ağlarının yapısından bahsedecek olursak, yapay sinir hücreleri bir araya gelerek yapay sinir ağını oluşturmaktadır. Sinir hücrelerinin bir araya gelmesi rastgele olmaz. Genel olarak hücreler 3 katman halinde ve her katman içinde paralel olarak bir araya gelerek ağı oluştururlar.

****

* 1. **Önerilen yöntem: Meta-sezgisel boyut azaltma ve sınıflandırma algoritması**

Proje de öncelikle sisteme yüklenen veri seti (problem uzayını homojen bir şekilde temsil eden ve yeterince sayıda örnekten oluşan) okunup, problem parametreleri bağımlı ve bağımsız değişkenler olarak etiketlenecektir.

Problemin tüm bağımsız değişkenlerinin yer aldığı modelin performans ölçülümü yapılacaktır. Performans ölçümü yaparken yapay sinir ağları kullanılarak performans eşik değeri bulunur. Yapay sinir ağları algoritmasının uygulanması sonucu ortaya çıkan performans eşik değerine hata tolerans değeri (hatanın ne kadar tolere edilebileceğine dair bir değer) eklenerek modelin eşik hata büyüklüğü bulunur.

Problem boyutu azaltma ve öznitelik çıkarımı aşamasına geçildiğinde eşik hata büyüklüğü değerine bakılarak performans karşılaştırılması yapılacaktır. Önerilen (boyutu azaltılmış, özellik çıkarımı yapılmış) modelin performansı ölçülüp klasik (problem boyutu azaltılmamış) modele göre performansında bir azalış yoksa veya tolere edilebilecek düzeyde bir azalış varsa başarı sağlanmış olacaktır.

* + 1. **Problem boyutu azaltma ve öznitelik seçimi**

1. Sezgisel k-nn ile ön filtreleme aşaması (özellik çıkarımı):

Probleme ait etiketlendirilmiş niteliklerden oluşan veri seti kullanılarak k-nn algoritması uygulanacaktır. K-nn algoritmasının kullanılma amacı k-nn algoritmasının k-değerinin bir sonraki adımda kullanılacak olan melez bir tahmin ve sınıflandırma algoritması olan sezgisel k-nn algoritmasında kullanılacak olmasıdır.

Bağımsız değişkenlerin her birinin probleme aynı derecede etki etmemesinden dolayı sezgisel k-nn algoritması ile probleme ait bağımsız değişkenlerin ağırlıklandırılması sağlanacaktır. Niteliklerin probleme etkisi incelenip, probleme en az etki eden nitelikler teşhis edilecektir. Elde edilen sonuçlar neticesinde, eşik ağırlık değerinden (örneğin 0-1 arasında ağırlıklandırılan nitelikler için eşik ağırlık değeri 0,01 olarak alınabilir) sayısal olarak küçük ağırlık değerine sahip olan niteliklerin çıkartılması sağlanacaktır. Bu eşik ağırlık değeri kullanıcı arayüzünden değiştirilebilir olacaktır (belirli aralıklarda kalmak şartıyla). Böylelikle ön filtreleme aşaması gerçekleştirilmiş olur.

Filtrelenmiş problem nitelikleri çıkartıldıktan sonra kalan bağımsız değişkenlerle problemin performans ölçümü yapılacaktır. Performans ölçümü yapılırken sezgisel k-nn tahminleyicisi kullanılır. Algoritma da sezgisel k-nn algoritmasında da kullanılan k-nn algoritmasının ideal k değeri kullanılır.

1. Ortak yaşayan organizmalar algoritması ile filtreleme aşaması:

Ön filtreleme aşamasından sonra meta-sezgisel optimizasyon algoritmalarından biri olan ortak yaşayan organizmalar (SOS) algoritması ile filtreleme aşamasına geçilir.

SOS algoritması diğer popülasyon tabanlı algoritmalara benzer olarak problem uzayındaki genel en iyi çözümü bulmak için popülasyon-tabanlı bir iteratif yöntemdir. SOS algoritmasında popülasyona verilen isim ekosistemdir. Popülasyonun ilk oluşturulması aşamasında arama uzayında bir grup organizma rastgele olarak başlatılır. Her bir organizma problem için bir aday çözümü temsil etmektedir.

Bu projede her bir çözüm adayı araştırılan yani azaltılmış/düşürülmüş güncel problem boyutudur.

Problem boyutu azaltma problemi ayrık değerlidir. O yüzden SOS algoritmasının operatörlerinin (gerek popülasyon oluşturma gerekse de arama operatörlerinin) ayrık değerli probleme uygun olarak çalışabilmesi için düzenlenmesi gerekir. Çözüm adaylarının her bir boyutu problemin bir niteliğine karşılık gelmektedir.

i. Popülasyonun oluşturulması

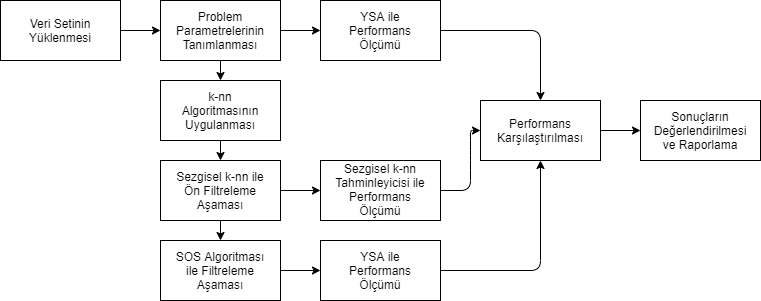
Başlangıç popülasyonunun yaratılmasında ayrık değerli probleme göre düzenleme yapılır. Problemdeki nitelik sayısı arasında rastgele tam sayılar üretilir. Üretilecek sayı adedi (yani çözüm adayı tasarım değişkeni sayısı) toplamda güncel problem boyutu kadar olacaktır. Çözüm adaylarının her biri ile YSA regresyon modeli oluşturulur. Amaç fonksiyon burada YSA’dır. Çözüm adayını YSA fonksiyonuna parametre olarak gönderip YSA’nın hata değeri elde edilecektir. Çözüm adayının YSA’ya parametre olarak gönderilmesi demek bu çözüm adayının niteliklerini içeren veri setinin (eğitim/doğrulama/test olarak YSA Matlab Tool’da üç alt sete ayrılıyor) YSA’ya giriş ve çıkış vektörleri olarak verilmesi demektir. Bu veri setini kullanarak eğitilen YSA modelinin hatası dikkate alınacaktır. Dolayısıyla bu bir minimizasyon problemidir.

ii. Arama süreci yaşam döngüsü

Burada SOS algoritmasının arama operatörlerinin sonuçları yuvarlanacak, ayrık değerli probleme uygun olarak bu operatörler tasarlanacaktır.  Sonlandırma kriteri olarak fitness fonksiyonunun azami çağırma sayısı veya hata eşik değerinin karşılanması dikkate alınacaktır. Problemin bağımsız değişkenlerinin kombinasyonlarından n-adet model oluşturulur.  N-adet modelin (popülasyondaki her bir çözüm adayının) her biri bahsedilen yapay zekâ/makine öğrenme tekniği kullanılarak çözümlenir. Çözümleme aşamasında yapay zekâ algoritmalarının test ve doğrulama süreçleri takip edilir. Ardından n-adet kombinasyonun performansları YSA ile ölçülür. Her bir modelin performansı “eşik hata büyüklüğü” ile karşılaştırılır. Eşik değerden yüksek performanslar varsa içlerinde en düşük boyutlu olanı seçilir. Aksi

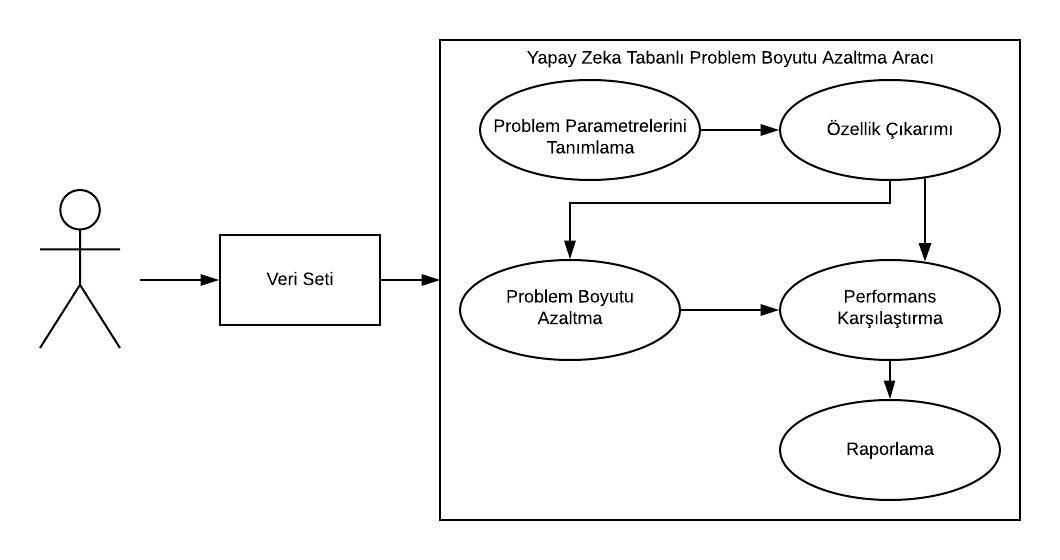
halde eşik değere yakın ya da kabul edilebilir bir performansa sahip olan modeller içerisinden seçim yapılır. İkinci seçenekte gerçekleşmediyse problem boyutu azaltma işlemi gerçekleştirilmez.

* + 1. **Sistemin blok diyagramı**



Şekil 3. Sistemin blok diyagramı

* + 1. **Kullanım senaryosu diyagramı**



Şekil 4. Kullanım senaryosu diyagramı

1. **PROJE YÖNETİMİ** 
   1. **İş- Zaman Çizelgesi**

**İŞ-ZAMAN ÇİZELGESİ**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **İP No** | **İş Paketlerinin Adı ve Hedefleri** | **Kim(ler) Tarafından Gerçekleştirileceği** | **Zaman Aralığı**  **(..-.. Ay)** | **Başarı Ölçütü ve Projenin Başarısına Katkısı** |
| 1 | Literatür Araştırması, Problem tanımı ve planlama | Büşra Aras ve Orhun Yıldız | 1 Temmuz - 10 Ağustos (2019) | Fikir edinebilmek için başka kaynaklardan literatür araştırmasının yapılması ve geliştirilecek sistemin fonksiyonel ve fonksiyonel olmayan gereksinimlerine kaynaklık teşkil edecek bilgilerin tanımlanması: Önerilen yöntemin tatbik edilebildiği ve problem boyutunun azaltılmasıyla ilgili bir modelin fonksiyonel ve fonksiyonel olmayan gereksinimleriyle birlikte tanımlanması. Katkı: %10 |
| 2 | Algoritma Analizi | Büşra Aras ve Orhun Yıldız | 11 Ağustos - 15 Eylül (2019) | Algoritmanın ihtiyaçlarının belirlenmesi ve bir yol haritasının çıkarılması. Katkı: %10 |
| 3 | Algoritma Tasarımı | Büşra Aras ve Orhun Yıldız | 16 Eylül - 25 Ekim (2019) | Kullanım senaryolarının ve sınıf diyagramlarının oluşturulması: 4+1 UML diyagramlarının hazırlanması. Katkı: %10 |
| 4 | Algoritmanın Kodlanması | Büşra Aras ve Orhun Yıldız | 17 Kasım (2019) - 21 Mart (2020) | Problem boyutunun azaltılmasıyla ilgili önerilen yöntemin geliştirilmesi. Katkı: %25 |
| 5 | Web Tabanlı Uygulamanın Geliştirilmesi ve Entegrasyon Çalışmaları | Büşra Aras ve Orhun Yıldız | 22 Mart - 25 Nisan (2020) | Araştırmacılar tarafından uygulama ücretsiz bir şekilde erişim sağlanabilmesi için web ortamının geliştirmesi ve entegrasyon çalışmalarının gerçekleştirilmesi. Katkı: %15 |
| 6 | Test, Doğrulama ve Bakım Çalışmaları | Büşra Aras ve Orhun Yıldız | 26 Nisan - 17 Mayıs (2020) | Araştırmacıların problemlerine ait veri setlerini web ortamına yükleyerek meta-sezgisel boyut azaltma ve sınıflandırma yöntemiyle problem boyutunun azaltılması ve sonuçları diğer yöntemlerle kıyaslayabilmesi. Katkı: %30 |

* 1. **Risk Yönetimi**

**RİSK YÖNETİMİ TABLOSU**

|  |  |  |
| --- | --- | --- |
| **İP No** | **En Önemli Riskler** | **Risk Yönetimi (B Planı)** |
| 1 | Zaman kısıtından dolayı projeyi yetiştirememe | Öncelikle SOS, K-nn, YSA ve önerilen algoritma geliştirilecektir ve kullanıma sunulacaktır. Önerilen algoritma ve problem boyutunun azaltılması için kullanılan diğer yöntemler arasında bir kıyaslama yapılacaktır. Bunun yanında kodlama aşamasında etkili bir görev paylaşımı yapılarak iş yükü azaltılacaktır ve projenin tüm fonksiyonlarıyla çalışması sağlanacaktır. |
| 2 | Önerilen algoritmanın beklendiği kadar başarılı olmaması | Bu durumda farklı meta-sezgisel arama algoritmalar kullanılarak önerilen yöntem ortaya çıkacaktır. Ayrıca problem boyutunun azaltılması ile ilgili kullanılan diğer yöntemlere alternatif bir yöntem sunabilmek de değerlidir. |

* 1. **Araştırma Olanakları**

**ARAŞTIRMA OLANAKLARI TABLOSU**

|  |  |
| --- | --- |
| **Kuruluşta Bulunan Altyapı/Ekipman Türü, Modeli**  (Laboratuvar, Araç, Makine-Teçhizat, vb.) | **Projede Kullanım Amacı** |
| Bilgisayar laboratuvarları | Geliştirilecek olan algoritmanın test ve doğrulaması için kullanılacaktır. |

1. **YAYGIN ETKİ**

**ARAŞTIRMA ÖNERİSİNDEN BEKLENEN YAYGIN ETKİ TABLOSU**

|  |  |
| --- | --- |
| **Yaygın Etki Türleri** | **Önerilen Araştırmadan Beklenen Çıktı, Sonuç ve Etkiler** |
| **Bilimsel/Akademik**  (Makale, Bildiri, Kitap Bölümü, Kitap) | Proje çerçevesinde 1 adet bildiri ve 1 adet makale çalışması yapılacaktır. |
| **Ekonomik/Ticari/Sosyal**  (Ürün, Prototip, Patent, Faydalı Model, Üretim İzni, Çeşit Tescili, Spin-off/Start- up Şirket, Görsel/İşitsel Arşiv, Envanter/Veri Tabanı/Belgeleme Üretimi, Telife Konu Olan Eser, Medyada Yer Alma, Fuar, Proje Pazarı, Çalıştay, Eğitim vb. Bilimsel Etkinlik, Proje Sonuçlarını Kullanacak Kurum/Kuruluş, vb. diğer yaygın etkiler) | Araştırmacılar, proje çerçevesinde geliştirilecek olan ürüne web ortamından erişim sağlayabilecek ve problem boyutunun azaltımı noktasında çalışmaları adına destek alabilecekler. |
| **Araştırmacı Yetiştirilmesi ve Yeni Proje(ler) Oluşturma**  (Yüksek Lisans/Doktora Tezi, Ulusal/Uluslararası Yeni Proje) | Proje konusunu araştırırken çok sayıda uluslararası makaleye rastladık. Dolayısıyla projenin konusu lisansüstü çalışmaların yapılabileceği niteliktedir. Lisans sonrasında lisansüstü eğitimimize devam etme planımız bulunmaktadır. Bu projeyi başarıyla tamamlamamız ve bu konuda danışmanımızla birlikte makale hazırlamamız durumunda lisansüstü eğitim ve yeni projeler yapabilme konusunda da ciddi bir yol almamız söz konusu olabilecektir. Genel olarak bu projeyi başarılı bir şekilde tamamladıktan sonra proje yapabilme tecrübesi ve birçok konuda yeterlilik kazanmış olacağız. |

1. **DİĞER KONULAR**

Proje çerçevesinde geliştirilecek olan uygulamanın test ve doğrulanmasında uluslararası bir veri havuzu paylaşım uygulaması olan “UCI Machine Learning Data Repository” kullanılacaktır. Bu uygulamaya aşağıdaki bağlantıdan erişilebilmektedir:

<https://archive.ics.uci.edu/ml/index.php>

1. **EKLER**

**EK-1: KAYNAKLAR**

1. He, B., Shah, S., Crystal Maung, G. A., Wan, G., & Schweitzer, H. (2019). Heuristic Search Algorithm for Dimensionality Reduction Optimally Combining Feature Selection and Feature Extraction.
2. Wang, F., & Sun, J. (2015). Survey on distance metric learning and dimensionality reduction in data mining. *Data mining and knowledge discovery*, *29*(2), 534-564.
3. Bingham, E., & Mannila, H. (2001, August). Random projection in dimensionality reduction: applications to image and text data. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 245-250). ACM.
4. Houari, R., Bounceur, A., Kechadi, M. T., Tari, A. K., & Euler, R. (2016). Dimensionality reduction in data mining: A Copula approach. *Expert Systems with Applications*, *64*, 247-260.
5. Azar, A. T., & Hassanien, A. E. (2015). Dimensionality reduction of medical big data using neural-fuzzy classifier. *Soft computing*, *19*(4), 1115-1127.
6. Zhong, X., & Enke, D. (2017). Forecasting daily stock market return using dimensionality reduction. *Expert Systems with Applications*, *67*, 126-139.
7. Padmaja, D. L., & Vishnuvardhan, B. (2016, February). Comparative study of feature subset selection methods for dimensionality reduction on scientific data. In *2016 IEEE 6th International Conference on Advanced Computing (IACC)* (pp. 31-34). IEEE.
8. Almusallam, N. Y., Tari, Z., Bertok, P., & Zomaya, A. Y. (2017). Dimensionality reduction for intrusion detection systems in multi-data streams—A review and proposal of unsupervised feature selection scheme. In *Emergent Computation* (pp. 467-487). Springer, Cham.
9. Guo, B., Hou, C., Nie, F., & Yi, D. (2016, December). Semi-supervised multi-label dimensionality reduction. In *2016 IEEE 16th International Conference on Data Mining (ICDM)* (pp. 919-924). IEEE.
10. Vasan, K. K., & Surendiran, B. (2016). Dimensionality reduction using principal component analysis for network intrusion detection. Perspectives in Science, 8, 510-512.
11. Yi, S., Lai, Z., He, Z., Cheung, Y. M., & Liu, Y. (2017). Joint sparse principal component analysis. Pattern Recognition, 61, 524-536.
12. Pierson, E., & Yau, C. (2015). ZIFA: Dimensionality reduction for zero-inflated single-cell gene expression analysis. Genome biology, 16(1), 241.
13. Jolliffe, I. T., & Cadima, J. (2016). Principal component analysis: a review and recent developments. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 374(2065), 20150202.
14. Rodríguez, R., Mexicano, A., Bila, J., Cervantes, S., & Ponce, R. (2015). Feature extraction of electrocardiogram signals by applying adaptive threshold and principal component analysis. Journal of applied research and technology, 13(2), 261-269.
15. Ahmed, S., Mafarja, M., Faris, H., & Aljarah, I. (2018, March). Feature selection using salp swarm algorithm with chaos. In Proceedings of the 2nd International Conference on Intelligent Systems, Metaheuristics & Swarm Intelligence (pp. 65-69). ACM.
16. Ahmed, S., Mafarja, M., Faris, H., & Aljarah, I. (2018, March). Feature selection using salp swarm algorithm with chaos. In Proceedings of the 2nd International Conference on Intelligent Systems, Metaheuristics & Swarm Intelligence (pp. 65-69). ACM.
17. Hegazy, A. E., Makhlouf, M. A., & El-Tawel, G. S. (2018). Dimensionality reduction using an improved whale optimization algorithm for data classification. International Journal of Modern Education and Computer Science, 10(7), 37.
18. Barani, F., Mirhosseini, M., & Nezamabadi-Pour, H. (2017). Application of binary quantum-inspired gravitational search algorithm in feature subset selection. Applied Intelligence, 47(2), 304-318.
19. Kotekar, S., & Kamath, S. S. (2018). Enhancing Web Service Discovery Using Meta-heuristic CSO and PCA Based Clustering. In Progress in Intelligent Computing Techniques: Theory, Practice, and Applications (pp. 393-403). Springer, Singapore.
20. Cheng, M. Y., & Prayogo, D. (2014). Symbiotic organisms search: a new metaheuristic optimization algorithm. Computers & Structures, 139, 98-112.
21. Arslan, F., & KAHRAMAN, H. T. Yapay Zekâ Tabanlı Büyük Veri Yönetim Aracı. Journal of Investigations on Engineering and Technology, 2(1), 8-21.
22. 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.
23. 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.
24. Yilmaz, C., Kahraman, H. T., & Söyler, S. (2018). Passive mine detection and classification method based on hybrid model. IEEE Access, 6, 47870-47888.
25. Bayindir, R., Colak, I., Sagiroglu, S., & Kahraman, H. T. (2012, December). Application of adaptive artificial neural network method to model the excitation currents of synchronous motors. In 2012 11th International Conference on Machine Learning and Applications (Vol. 2, pp. 498-502). IEEE.
26. Kahraman, H. T., Bayindir, R., & Sagiroglu, 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.
27. Yeşilbudak, M., Kahraman, H., & Karacan, H. (2011). Veri madenciliğinde nesne yönelimli birleştirici hiyerarşik kümeleme modeli. Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi, 26(1).
28. Adak, M. F., & Yurtay, N. (2013). Gini Algoritmasını Kullanarak Karar Ağacı Oluşturmayı Sağlayan Bir Yazılımın Geliştirilmesi. Bilişim Teknolojileri Dergisi, 6(3), 1-6.
29. Lezki, Ş. (2014). Çok Kriterli Karar Verme Problemlerinde Karar Ağacı Kullanımı. İktisadi Yenilik Dergisi, 2(1), 16-31.
30. Tokgöz, B. (2017). “Iş Gereksinimi Odaklı Test Senaryoları Üretim Modeli”, Necmettin Erbakan Üniversitesi Fen Bilimleri Enstitüsü Yüksek Lisans Tezi, 8-13.
31. Birbil, İ., "Tahmin ve Çıkarım 4 - Boyut Küçültme", <http://www.veridefteri.com/2018/06/19/tahmin-ve-cikarim-4-boyut-kucultme/> (Son erişim tarihi: 20.08. 2019)
32. Şimşek, H.K., "Boyut Azaltma: Temel Bileşen Analizi", [https://medium.com/deep-learning-turkiye/boyut-azaltma-temel-bileşen-analizi-812fd2163bbf](https://medium.com/deep-learning-turkiye/boyut-azaltma-temel-bile%C5%9Fen-analizi-812fd2163bbf) (Son erişim tarihi: 20.08. 2019)
33. Şeker, Ş.A., "Özellik Çıkarımı (Feature Extraction)", http://bilgisayarkavramlari. sadievrenseker.com/2008/12/01/ozellik-cikarimi-feature-extraction/ (Son erişim tarihi: 20.08. 2019)
34. Pala, T., Yücedağ, İ., Kahraman, H. T., Güvenç, U., & Sönmez, Y. (2018, September). Haar Wavelet Neural Network Model. In 2018 International Conference on Artificial Intelligence and Data Processing (IDAP) (pp. 1-8). IEEE.
35. Holland, J.H., 1975. "Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence". Q. Rev. Biol. 1, 211. http://dx.doi.org/10.1086/418447.
36. Kahraman, H. T., Aras, S., Sönmez, Y., Güvenç, U., & Gedikli, E. Analysis, Test and Management of the Meta-Heuristic Searching Process: An Experimental Study on SOS. Politeknik Dergisi.
37. Aras, S., Kahraman, H. T., Sönmez, Y., & Güvenç, U. Meta-Sezgisel Arama Algoritmalarının Test Edilmesi İçin Yeni Yöntemlerin Geliştirilmesi. İleri Teknoloji Bilimleri Dergisi, 6(2), 1-8.
38. Murty, K. G.(2003). Optimization Models For Decision Making, vol. 1, Internet Edition, Chapter1: Models for Decision Making, 1-18, 2003.
39. Dosoglu, M. K., Guvenc, U., Duman, S., Sonmez, Y., & Kahraman, H. T. Symbiotic organisms search optimization algorithm for economic/emission dispatch problem in power systems. Neural Computing and Applications, 29(3), 721-737, 2018.
40. Baysal Y.A., Altaş İ.H., Gedikli E., "Simbiyotik Organizmalar Arama Algoritması ile Dağıtım Şebekelerinde Güç Kayıplarının Azaltılması", Akıllı Sistemlerde Yenilikler ve Uygulamaları Sempozyumu (ASYU 2016), DÜZCE, TÜRKIYE, 29 Eylül - 1 Ekim 2016, ss.295-299