



T.C.

SELÇUK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

FİNANS SEKTÖRÜ İÇİN YAPAY ÖĞRENME TEKNİKLERİ KULLANARAK KREDİ KULLANABİLİRLİĞİN TESPİTİ

Ali TUNÇ

YÜKSEK LİSANS

Bilgisayar Mühendisliği Anabilim Dalı

Kasım-2016
KONYA
Her Hakkı Saklıdır

ÖZET

YÜKSEK LİSANS TEZİ

FİNANS SEKTÖRÜ İÇİN YAPAY ÖĞRENME TEKNİKLERİ KULLANARAK KREDİ KULLANABİLİRLİĞİN TESPİTİ

Ali TUNÇ

Selçuk Üniversitesi Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Erkan ÜLKER

2016, 73 Sayfa

Jüri
Doç. Dr. Harun UĞUZ
Doç. Dr. Erkan ÜLKER
Yrd. Doç. Dr. Onur İNAN

Bilgi teknolojilerindeki gelişmelerle birlikte, banka şirketleri, müşterilerinin kredi taleplerini etkili analitik yöntemler ve risk analizleri ile değerlendirebilmektedirler. Kredi skorlama sistemleri olarak adlandırılan yazılım ürünleri genel olarak daha önce belirlenen kredi faktörlerine göre müşterinin verilerinin toplanması, elde edilen verinin çeşitli istatistiksel veya makine öğrenmesi teknikleriyle işlenmesi ve kredi risk analizinin yapılarak nihai kredi kararının belirlenmesi aşamalarından oluşur. Kredi kararı aşamalarında oluşabilecek hataları önlemeye yardımcı olan unsur, farklı karar faktörlerinin değerlendirilmesinde standart bir çözüm sunan otomatik skorlama araçları ve modellerinin geliştirilmesidir. Kredi skorlama sistemlerinde kullanılmakta olan hataya meyilli istatiksel analiz metodolojilerinin yerine, her bankanın kredi kriterlerine göre uyarlanabilecek, kesinliği yüksek makine öğrenmesi tekniklerinin sunulduğu bir çözüm üzerine çalışılmıştır. Bu çalışma, kredi risk faktörlerinin belirlenmesi, elde edilen verinin makine öğrenmesi algoritmaları ile işlenmesi ve veri tutarlığı ile oluşturulan tahminlerin analizi algoritmalarının geliştirilmesi aşamalarından oluşmaktadır.

Kredi başvurularının değerlendirilme sürecinde çeşitli skorlama modelleri yaygın olarak kullanılmaktadır. Bu modeller dâhilinde müşterilerin geçmiş banka hareketleri işlenerek kredi kararı verilebilmektedir. Yapılan çalışma ile müşteriye ait değişken kümelerinden oluşan veriler, makine öğrenmesi teknikleriyle işlenerek, müşteriye ait kredi değeri belirlenmeye çalışılmıştır. Bayes ve gri kurt optimizasyonu yöntemleri ile sınıflandırma problemi olarak çalışma odağı oluşturulmuştur. Elde edilen bu bilgilere göre müşteriye kredi verilebilir ya da verilemez kararı ortaya çıkarılmıştır.

Bu tezde kredi başvurusunda bulunan ve kredi kullanan tüketicilerdeki artışı sağlıklı yönetebilecek yapı ihtiyacının karşılanması, doğru müşteriye, doğru zamanda, doğru miktarda ve doğru vadede kredi vermeyi sağlayacak yapının kurulması, kredi tahsilatlarının verimliliğinin arttırılması, riski minimize ederek karlılığın maksimum noktaya getirecek optimum stratejilerin oluşturulması, bankanın kredi skorlamasında ve değerlendirme sisteminde uzman görüş etkisini azaltılması ve maliyetlerin düşürülmesi amaçlanmıştır.

Anahtar Kelimeler: Kredi Risk Analizi, Kredi Skor Modellemesi, Makine Öğrenmesi, Müşteri Segmentasyonu, Sınıflandırma Algoritmaları

ABSTRACT

MS THESIS

USING MACHINE LEARNING TECHNIQUES OF DETECT THE CREDIT AVAILABILITY FOR THE FINANCIAL SECTOR

Ali TUNÇ

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

Advisor: Doç. Dr. Erkan ÜLKER

2016, 73 Pages

Jury
Doç. Dr. Harun UĞUZ
Doç. Dr. Erkan ÜLKER
Yrd. Doç. Dr. Onur İNAN

With the advances in the Information Technology (IT) field, banks can evaluate the credit requests of the customers via effective analytical methods and risk analysis. The software products, named Credit Scoring Systems, consist of collecting customer data based on pre-determined credit factors, processing the data with various statistical or machine learning methods, and conducting credit risk analysis to make the final credit decision. In order to reduce the mistakes made while taking credit approval decisions, automatic scoring tools and models, offering a standard solution for evaluating different decision factors, should be developed. Instead of error-prone statistical analysis methodologies that are used in credit scoring systems, we studied on a new solution which utilizes machine learning techniques with high accuracy and can be customized for the credit criteria of each bank. This work consists of the following phases: determining the credit risk factors, processing the acquired customer data with machine learning algorithms, and developing analysis algorithms of the predictions made by the data consistency.

Throughout the evaluation process of the credit applications, various scoring models are commonly used. These models utilize the previous transactions on the bank accounts of the customers to make a decision on the credit applications. In the proposed work, the information about the customer related to several aspects and processed with machine learning techniques, and finally a credit score will be determined for each customer. Classification problem using Bayes and Grey Wolf optimization methods was focused in this work. This information will later be used to decide whether the credit application of a customer can be approved or not.

In this thesis, intentions can be summarized as, providing useful tools to manage the increasing number of customers who apply for consume credits, establishing a structure for crediting the right customers at the right time with the right amount and payment plan, increasing the efficiency of collecting credit payments, thus contributing to the national economy by using the resources more effectively, creating optimal strategies for maximizing the profit by minimizing the risk, reducing the effect of an expert for credit scoring and evaluation, and reducing the costs.

Keywords: Classification Algorithms, Credit Risk Analysis, Credit Scores Modeling, Customer Segmentation, Machine Learning

ÖNSÖZ

Yüksek lisans sürecimde bilimsel katkıları ile yardımlarını benden esirgemeyen, gösterdiği yakınlık, özveri ve samimiyetle çalışmalarımın her aşamasında danışmanlığı ile yol gösteren değerli hocam ve danışmanım Sayın Doç.Dr. Erkan ÜLKER'e teşekkürlerimi sunarım. Önerilerini ve bilgilerini benimle paylaşan tüm hocalarıma, yüksek lisans çalışmam boyunca manevi destekleri ile varlığını hissettiren ve her daim yanımda olan aileme sonsuz teşekkür ve şükranlarımı sunarım.

Ali TUNÇ
KONYA-2016

İÇİNDEKİLER

ÖZET ..... **iv**

ABSTRACT ..... **v**

İÇİNDEKİLER ..... **vii**

SİMGELER VE KISALTMALAR ..... **ix**

1. GİRİŞ **1**

2. KAYNAK ARAŞTIRMASI **4**

2.1. Kredi Skorlama İle İlgili Literatür Çalışmaları 4
2.2. Özellik Seçimi İle İlgili Literatür Çalışmaları 9
2.3. Yapay Öğrenme Teknikleri İle İlgili Literatür Çalışmaları 10

3. VERİ SETİ VE VERİ SETİ ÜZERİNDEKİ İŞLEMLER **17**

3.1. Veri Setinin Oluşturulması 17
3.2. Veri Madenciliği ve Veri Ön İşleme Teknikleri 18
3.2.1. Verinin Temizlenmesi 20
3.2.2. Verinin Bütünleştirilmesi 20
3.2.3. Verinin İndirgemesi 20
3.2.4. Verinin Dönüştürülmesi 21
3.2.5. Veri Madenciliği Algoritmalarının Uygulanması 21
3.2.6. Sonuçlar ve Değerlendirmeler 21
3.3 Normalizasyon 22
3.3.1. Ondalık Ölçekleme 22
3.3.2. Min-Max Normalleştirme 22
3.3.3. Z-Score Standartlaştırma 23
3.4 Veri Düzeltme 23
3.4.1. Veri Düzeltme İçin Veri Gruplama Metodu (Binning Methods) 23
3.4.2. Beş Sayı Özeti Metodu (The Five Number Summary Metod) 25

4. OPTİMİZASYON ALGORİTMALARI **28**

4.1 ÖZELLİK SEÇİMİ VE ÖZELLİK SEÇİMİ ALGORİTMALARI 28
4.1.1. Bilgi Kazanımı (Information Gain) Algoritması 28
4.1.2. Kazanım Oranı (Gain Ratio) Algoritması 30
4.2 SINIFLANDIRMA ALGORİTMALARI 32
4.2.1 GRİ KURT OPTİMİZASYONU (GWO) 32
4.2.1.1. Sosyal Hiyerarşi 34
4.2.1.2. Avı Çevreleme 34
4.2.1.3. Avlanma 35
4.2.1.4. Ava Saldırma (Sömürü) 36
4.2.1.5. Av Arama (Keşif) 37
4.2.2 BAYES ALGORİTMASI 38
4.2.2.1. Naive Bayes Sınıflandırma 41

4.2.2.2. Dynamic Bayes Ağı 42

5. ARAŞTIRMA SONUÇLARI VE TARTIŞMA 43

5.1. Çalışma Ortamı ve Uygulamanın Geliştirilmesi 44
5.2. Sınıflandırma Sonuçları 48

6. SONUÇLAR VE ÖNERİLER 57

6.1 Sonuçlar 57
6.2 Öneriler 59

KAYNAKLAR 60

EKLER 67

ÖZGEÇMİŞ 72

SİMGELER VE KISALTMALAR

Kısaltmalar

LR **:** Logistic Regression
YSA **:** Yapay Sinir Ağları
NB **:** Naive Bayes Algoritm
DB **:** Dynamic Bayesian Algoritm
GS **:** Genetic Search Algoritm
SOM **:** Self-Organizing Maps Algoritm
DVM **:** Destek Vektör Makinaları
A* : A* (A Star) Araması
BS **:** Beam Search
HC **:** Hill Climbing
BFS **:** Best First Search
PSO **:** Parçacık Sürü Optimizasyonu
ACO **:** Ant Colony Optimization
ABC **:** Artificial Bee Colony
GWO **:** Gry Wolf Optimization
SSO **:** Social Spider Optimization
Min **:** Minimum
Max **:** Maximum
ÖS : Özellik Seçimi
İPSO : İkili Parçacık Sürü Optimizasyonu
KKO : Karınca Koloni Optimizasyonu
GSA : Gravity Search Algorithm
DEA : Differential Evolution Algorithm

1. GİRİŞ

Bilgi teknolojilerindeki gelişmelerle birlikte, banka şirketleri, müşterilerinin kredi taleplerini etkili analitik yöntemler ve risk analizleri ile değerlendirebilmektedirler. Kredi skorlama sistemleri olarak adlandırılan yazılım ürünleri genel olarak daha önce belirlenen kredi faktörlerine göre müşterinin verilerinin toplanması, elde edilen verinin çeşitli istatistiksel veya makine öğrenmesi teknikleriyle işlenmesi ve kredi risk analizinin yapılarak nihai kredi kararının belirlenmesi aşamalarından oluşur. Bu sistemlerde amaç kredi kararı aşamalarında oluşabilecek hataları önlemeye yardımcı olan ve farklı karar faktörlerinin değerlendirilmesinde standart bir çözüm sunan modellerin geliştirilmesidir. Kredi skorlama sistemlerinde kullanılmakta olan istatiksel analiz metodolojilerinin yerine, her bankanın kredi kriterlerine göre uyarlanabilecek, kesinliği yüksek yapay öğrenme tekniklerinin kullanıldığı bir çözüm üzerinde çalışma yapılmıştır.

Kredi skorlama sistemleri; müşterilerin kredi marketine erişim imkânını arttırması, kredi ücretlendirmesinin düşürülmesi, karar süreçlerinin kısaltılması ve karar aşamalarına bağlı ortaya çıkan hata ile temerrütlerin azaltılmasında önemli bir rol oynayarak ülkenin ekonomik gelişimine katkı sağlamaktadır. Yapılan çalışma ile bankaların tüm kredi kararlarını otomatik bir skorlama sistemi üzerinden vermesi, oldukça maliyetli olan banka yetkilisinin kredi sonucunu belirlediği yöntemlere göre daha az maliyetli, hızlı ve kesinliği yüksek bir sistemin oluştura bilirliği incelenmiştir.

Yapılan saha araştırmalarında, makine öğrenmesine dayalı kredi skorlama sistemleri tüm dünyada büyüyen bir pazar olarak görülmektedir. Türkiye Merkez Bankası'nın yaptığı bir araştırma dâhilinde, 2007 yılının ilk çeyreğinde %5 düzeyinde seyreden tüketici kredisi miktarının son yıllarda %20 seviyelerine yükseldiği belirlenmiştir. Bu istatistiksel veriler, banka kredilerine duyulan talepteki artışın ve buna bağlı olarak geliştirilen efektif kredi karar modellerinin oluşturulması ihtiyacının somut bir göstergesidir. Bu pazarda başarılı olabilmek için değişen ve gelişen ihtiyaçlara cevap veren efektif, kesinlik oranı yüksek karar mekanizmalarına sahip olmak gerekmektedir.

Bununla birlikte, geliştirilen yeni karar modeli ile elde edilmek istenen en büyük katkı, müşterilere ait yeterli verinin bulunmadığı belirsiz koşullarda ortaya etkin bir sonucun çıkarılmaya çalışılmasıdır. Geliştirilen çalışma ile ulusal ekonomiye en büyük katkıları, kredi tahsisatlarının verimliliğinin artırılması ve buna bağlı olarak kaynakların

en üretken uygulamalara yönlendirilmesi, üretkenliğin ve ekonomideki büyümenin artmasının sağlanmasıdır.

Diğer yandan, geliştirilen sistemden finansal kuruluşlar da ciddi faydalar elde edeceklerdir. Kredi karar sistemlerindeki iyileştirmeler, yani doğru müşteriye doğru zamanda, doğru miktarda ve doğru vadede kredi vermeyi sağlamak gibi şirketlerin hedeflediği market üzerinde üstünlük elde edebilmesine, kar oranını ve varlığını sürdürebilme olasılığını artırabilmesine olanak sağlamaktadır.

Yapılan çalışma ile bankaların tüm kredi kararlarını otomatik bir skorlama sistemi üzerinden vermesi, hataya meyilli ve oldukça maliyetli olan banka yetkilisinin kredi sonucunu belirlediği yöntemlere göre daha az maliyetli, hızlı ve kesinliği yüksek olacaktır.

Çalışma sayesinde şirketler hem mevcuttaki temel veri modelleri hem de kredi risk analizi yazılımı alanında yapacağı köklü değişikliklerle, sektöre daha profesyonel bir sistem sunmayı istemektedir. Bu alanlarda yapılan geliştirmelerle banka şirketlerinin istatistiksel skorlama modellerinden kaynaklanan kısıtlamalara maruz kalmadan kredi kararlarını verebilmeleri sağlanması hedeflenmiştir. Geliştirilen makine öğrenmesi tabanlı yeni teknolojinin kesinliği yüksek kredi skorlama çözümleri üretilmesinde ciddi faydalar sağlayacağı öngörülmektedir. Bu anlamda, kredi tahsisinde oluşabilecek hataların önlenmesiyle birlikte, ekonomik büyümede yatırım amaçlı kullanılan anaparanın artışını sağlamak amaçlanmıştır.

Tezin genel amacı; kredi kararı aşamalarında oluşabilecek hataları önlemeye yardımcı olan, farklı karar faktörlerinin değerlendirilmesinde standart bir çözüm sunan otomatik skorlama araçları ve modellerinin geliştirilmesidir. Kredi skorlama sistemlerinde kullanılmakta olan hataya meyilli istatiksel analiz ile yapay öğrenme (makine öğrenmesi) teknikleri kullanımı sayesinde sonuçların daha kararlı ve doğru çıkacağı düşünülmüştür.

Kredi başvurularının değerlendirilme sürecinde çeşitli skorlama modelleri yaygın olarak kullanılmaktadır. Bu modellerde müşterilerin geçmiş banka hareketleri işlenerek kredi kararı verilebilmektedir. Bu çalışma, kredi risk analizi üzerinde etkili olan faktörlerin belirlenmesi ve tutarlı bir analizinin yapılabilmesi amacıyla geliştirilen sistematik bir metodolojiye dayanmaktadır. Geliştirilen skorlama modeliyle toplanan

olan müşteriye ait değişken kümelerinden oluşan veriler, yapay öğrenme (makine öğrenmesi) teknikleriyle işlenerek, müşteriye kredi verilebilir ya da verilemez kararı ortaya çıkarılmıştır.

Bu sistemlerle yapay öğrenme (makine öğrenmesi) tabanlı yeni teknolojinin kesinliği yüksek kredi skorlama çözümleri üretilmesinde ciddi faydalar sağlayacağı öngörülmektedir. Bu anlamda, kredi tahsisinde oluşabilecek hataların önlenmesiyle birlikte, ekonomik büyümede yatırım amaçlı kullanılan anaparanın artışını sağlamak amaçlanmıştır.

2. KAYNAK ARAŞTIRMASI

Bu bölümde kredi kullandırma, skor kartı modelleri geliştirme, müşteri segmentasyonu gibi süreçler tanıtılarak bu alanda yapılmış çalışmalara ait literatür taramaları sunulmuştur. Ayrıca özellik seçimi ve yapay öğrenme teknikleri ile ilgili gerekli literatür taraması yapılmıştır. Yapılan araştırmaların tamamı aşağıda sunulmuştur.

2.1. Kredi Skorlama İle İlgili Literatür Çalışmaları

Kredi skorlama bireyin kredi değerliliğinin sayısal ifadesidir. Genel hedef bireyin kredi puanını belirlemektir. Bir bireye verilecek tutar ve geri ödeme vadesi, kredi skorlama sürecinde belirlenir. Kredi skorlama, kredi geçmişi gibi belirli kriterlere bakar. Bunlar sayesinde bankalar ve mikro kredi kurumları gibi finans kurumlarının riskini genel varsayılan oranına göre azaltma niyeti ile yapılır.

Yapay zekâ teknikleri ve istatistiksel tabanlı yöntemler kullanılarak çeşitli kredi puanlama modelleri geliştirilmiştir. Kredi skorunu etkileyen faktörleri sıralarsak; aktif getirisi, alınan krediler, bölge, cinsiyet, kanuni takip durumu, kredi notu, medeni durum, meslek ve kıdem, ret edilen krediler ve teminat gibi değişkenler kredi skorunu etkileyen faktörlerdir. Kredi skorlama ile ilgili yapılan bazı literatür çalışmaları Çizelge 2.1.'de sunulmuştur.

Çizelge 2.1. Kredi Skorlama alanında yapılmış bazı çalışmalar

Yazar	**Yıl**	**Çalışma**
A. Lahsasna, R. Ainon, T. Wah	2009	Yazılımı geliştirerek kredi skorlama modeli çalışması yapmışlardır.
Mehmet Yazıcı	2011	Bankacılıkta kredi tahsisi çalışması ve kredi skorlamaya yönelik çalışmalar yapmıştır.
P. Yap, S. Ong, N. Husain	2011	Kredi skorlama modelleri üzerinde veri madenciliği algoritmaları kullanarak kredi değerlendirme çalışmaları yapmışlardır.
Thabiso Peter Mpofu ve Macdonald Mukosera	2012	Credit Scoring Techniques: A Survey üzerine çalışma yapmışlardır.
A. Heiat	2012	Bilgisayardan kredi skorlama için veri madenciliği üzerine çalışma yapmıştır.
Olsan, Delen ve Meng	2012	Kredi skorlama modellerinin geliştirilmesindeki veri bağımlılığı etkisini en aza indirmek için K-kat çapraz doğrulama kullanmışlardır.
Soner Akkoç	2012	Yapay Sinir Ağları ve Doğrusal Ayırma Analizi ile Kredi Derecelendirme üzerine çalışma yapmıştır.
Ercan Öztemel	2012	Yapay Sinir Ağları kitabında kredi skorlamaya yönelik örnek çalışma yapmıştır.
A. Blanco, R. Mejias, J. Lara, S. Rayo	2013	Mikro finans endüstrisi için kredi skorlama modelleri sinir ağları kullanılarak puanlama sistemi geliştirmiştir.
Hussain Ali Bekhet ve Shorouq Fathi Kamel Eletter	2014	Ürdün ticari bankalar için kredi riski değerlendirme modeli: Sinir puanlama yaklaşımı üzerine çalışmalar yapmıştır.
Ferdi Sönmez	2015	Kredi Skorunun Belirlenmesinde Yapay Sinir Ağları ve Karar Ağaçlarının Kullanımı: Bir Model Önerisi sunmuştur.

(Emel ve ark., 2003), ticari bankacılık sektörü için bir kredi skorlama yaklaşımı üzerine çalışmışlardır. Araştırmalarında müşterinin finansal performansını değerlendirmek için, skorlama yöntemlerini kullanarak kredilendirme puanlarının hesaplamaları üzerine çalışmışlardır.

(Shao ve ark., 2005), parçacık sürü optimizasyonu (PSO) ile sinir ağına dayalı kredi skorlama modeli üzerine çalışmışlardır. YSA üzerinde PSO algoritması kullanılarak optimizasyon sağlanmış ve PSO ile eğitim sürecinin yakınsamasını hızlandırma ve örüntü sınıflandırma doğruluğunun artırıldığı sonuçlarına ulaşmışlardır.

(Abdou, 2009), mısır bankalarında kredi skorlama modellerinin uygulanabilirliği ile ilgili çalışma yapmıştır. Mısırlı kamu bankalarının kredi skorlama modellerinin

analizinde, genetik algoritmalar üzerinde çalışılmış Lojistik Regresyon (LR) ile Genetik Algoritma'yı (GA) karşılaştırmıştır.

(Leung ve ark., 2007), bir yapay bağışıklık sistemi algoritması kullanarak tüketici kredi puanlama sistemi üzerine çalışma yapmışlardır. Çalışmada doğal yapay zekâ tekniği ile bağışıklık sistem ismini verdikleri bir zekâ tekniği karşılaştırılarak ortaya çıkan sonuçları yorumlamışlardır.

(Giannetti ve ark., 2008), potansiyel başvuru sayısının artmasının, kredi onay prosedürünün otomatikleşmesini ve borçlunun finansal sağlığını denetleyen ileri tekniklerin gelişmesine yardımcı olduğuna dair çalışma yapmışlardır.

(Tsai ve Wu, 2008), iflas tahmin ve kredi puanlama için sinir ağlarını kullanarak deney yapmışlardır. Yapay zekâ ve makine öğrenme tekniklerini bu finansal karar verme problemlerini çözmek için kullanılmışlardır. Bu optimal karar ile üç sınıflandırıcı mimarilerde iyi olduğunu göstermeye çalışmışlardır.

(Hu, 2009), ulusal öğrenci kredileri için yapay zekâ teknolojisi kullanarak kişisel kredi derecelendirme çalışması yapmıştır. Yapay sinir ağı teknikleri kullanılarak üniversite öğrencisi hakkında kredi notu değerlendirilmesinde oldukça verimli sonuçlar ortaya çıkarmıştır.

(Bhaduri, 2009), yapay bağışıklık sistemi algoritmalarını kullanarak kredi puanlama üzerine karşılaştırmalı bir çalışma yapmıştır. Yapay bağışıklık sistemi algoritmaları ile diğer yöntemlerle karşılaştırılarak algoritmalar arası başarı sonuçlarını karşılaştırmaya çalışmıştır.

(Liu ve ark., 2009), kredi derecelendirme analizi için yapay sinir ağları(YSA) araştırması yapmışlardır. Backpropagation ve Levenberg-Marquardt algoritmaları kullanarak YSA üzerinde kredi derecelendirme çalışması yapmışlardır. Bu metotların kredi tahminleme de uygulanmasının yararlı bir yöntem olduğu ortaya koymuşlardır.

(Lahsasna ve ark., 2008), yazılım hesaplama yöntemlerini kullanarak kredi skorlama modeli çalışması yapmışlardır. Hibrit hesaplama yöntemini kullanarak akıllı bir kredi puanlama modeli önermektedirler.

(Kamalloo ve Abadeh, 2010), kredi puanlamada belirsiz kuralları ayıklamak için bir yapay bağışıklık sistemi geliştirmesi çalışmasını yapmışlardır. Yapılan çalışmada model doğru bulanık if-then kuralları ayıklamak için bulanık desen sınıflandırma ile birleştirilmiştir. Sonuçlar önerilen bağışıklık tabanlı sınıflandırma sisteminin kredi risklerini tespitinde doğru olduğunu göstermektedir.

(Yazici, 2011), bankacılıkta kredi tahsisi çalışmasında kredi skorlamaya yönelik çalışmalar yapmıştır. Discriminant analizi ve YSA kullanılarak müşterilere kredi kartlarının doğru verilmesi noktasında başarılı sonuçlar ortaya çıkarmıştır.

(Abdou ve Pointon, 2011), kredi puanlama alanında, istatistiksel teknikler ve değerlendirme kriterleri adında literatür araştırması yaparak 214 makale üzerinde tarama yapmış, istatistiksel olarak kredi skorlama ile ilgili genişçe bir çalışma sunmuştur.

(Van Gool ve ark., 2012), mikro finans için kredi skorlama isimli çalışmada lojistik regresyon yöntemi ile Bosna veri seti üzerinde çalışmalar yapmış yeni modeller ortaya koymuştur.

(Yap ve ark., 2011), kredi skorlama modellerinin aracılığıyla kredi değerlendirmesini artırmak için veri madenciliği kullanmışlardır. Veri madenciliği yöntemleri kullanarak doğru kredi skorlama yapılabilmesi ile ilgili çalışmalar ortaya koymuşlardır.

(Oztemel, 2016), yapay sinir ağları isimli kitabında kurduğu sinir ağı modeli ile kredi skorlamaya yönelik yaptığı örnek bir çalışmaya yer vermiştir.

(Akkoc, 2010), yapay sinir ağları(YSA) ile doğrusal ayırma analizi modellerini kullanarak kredi skorlama çalışması yapmıştır. YSA ile kredi tahminleme yapılabilmesi üzerinde başarılı sonuçlar elde etmiştir.

(Mpofu ve Mukosera, 2012), kredi skorlama tekniği olarak anket ile kredi derecelendirmesi yapmıştır. Yapay zekâ teknikleri ile istatistiksel yöntemleri karşılaştırmıştır.

(Crone ve Finlay, 2012), kredi puanlama örneği, örneklem büyüklüğü ve dengeleme ile deneysel bir çalışma yaparak kredi puanını ölçme çalışması yapmıştır. 20 örnekler ve 29 rebalanced örnek dağılımları arasında iki veri setleri üzerinde lojistik regresyon, diskriminant analizi, karar ağaçları ve yapay sinir ağları göreceli doğrulukları değerlendirilmesinde karşılaştırmalar yapmışlardır.

(Olson ve ark., 2012), güvenilir bir şekilde tahmin sağlamak ve kredi skorlama modellerinin geliştirilmesindeki veri bağımlılığı etkisini en aza indirmek için K-kat çapraz doğrulama kullanmıştır. İflas ve tahsis tahminleri için karar ağaçları ile sinir ağları ve destek vektörleri kıyaslamışlardır.

(Heiat, 2012), bilgisayardan kredi skorlama için veri madenciliği performans karşılaştırması çalışması yapmıştır.

(Marques ve ark., 2012), kredi puanlama evrimsel hesaplamanın uygulanması hakkında bir literatür çalışması yapmışlardır. Bagging and AdaBoost yöntemleri ile

random subspace and rotation forest seçim yöntemlerinin deneysel sonuçları üzerine çalışılmışlardır.

(Fogarty, 2012), kredi puanlama sistemi koruma fonksiyonları için genetik algoritmalar tekniğini kullanmıştır. Genetik algoritmanın sonuç ve performansları üzerinde çalışmıştır. Geleneksel yöntemlere göre genetik algoritmanın daha iyi sonuçlar verdiği konusunda fikir bildirmiştir.

(Sadatrasoul ve ark., 2015), veri madenciliği teknikleri ile bankalar ve finans kurumlarında kredi puanlama alanında bir literatür taraması yapmışlardır. "sınıflandırma ve sınıflandırma" ve "kümelenme ve sınıflandırma" konularında incelemeler yapmışlardır.

(Blanco ve ark., 2013), mikro finans endüstrisi için kredi skorlama modelleri sinir ağları kullanılarak puanlama sistemi geliştirmişler ve bu geliştirmeleri Peru'da kanıtlanmışlardır. Sinir ağı modeli üzerine kurdukları yapının klasik tekniklere göre daha iyi performans gösterdiğini ortaya koymuşlardır.

(Baklouti, 2013), sınıflandırma ve regresyon ağacı üzerinden mikro finans kredi puanlamaya yönelik psikolojik yaklaşım çalışması yapmıştır. Gelecekteki varsayılan olayları tahmin ve borçluların psikolojik özellikleri rolünü araştıran çalışmada Tunuslu bir bankadan alınan mikro finans veriler üzerinde CART, lojistik regresyon ve diskriminant analiz tekniklerine göre modeller kurulmuş ve sonuç ve performanslarını göstermeye çalışmıştır.

(Bekhet ve Eletter, 2014), Ürdün ticari bankaları için kredi riski değerlendirme modeli geliştirmiştir ve buna sinir puanlama yaklaşımı adını vermişlerdir. Yapay sinir ağları, istatistiksel teknikler ve birçok alanda sınıflandırma problemlerinde başarılı sonuçlar almışlardır. Lojistik regresyon modeli, genel doğruluk oranı bakımından radyal tabanlı fonksiyon modeline göre biraz daha iyi bir performans göstermektedir. Ancak radyal temel işlevi varsayılan olabilecek yeni müşterilerin belirlenmesinde daha iyi sonuçlar çıkardığını göstermişlerdir.

(Sonmez, 2015), kredi skorunun belirlenmesinde yapay sinir ağları ve karar ağaçlarının kullanımı ile model önerisinde bulunmuştur. Bankalardan kredi talep eden bireysel müşterilerin taleplerinin değerlendirilerek başvurunun kabul ya da reddetme sonuç bilgisi için yapay sinir ağları (YSA) metodolojisini temel alan bir yazılım modeli önermiştir. Bir mevduat bankasına ait gerçek veri kümesi uygulamada kullanılmış ve sonuçları ayrıca geliştirilen karar ağacı (KA) modelinin sonuçları ile karşılaştırılmıştır. Bu iki modelde de bireysel kredi başvurusu için verilecek sonuç kararı numerik

değerlerden oluşan veriler üzerinden değerlendirilmektedir. Çalışmada ulaşılan sonuçlar, YSA modelinin müşteri kredi skorunun tespitinde yüksek öngörü doğruluğunu sağlama ve kredi riskini belirli ölçüde tahmin edebilmede KA modeline göre başarılı olduğunu göstermektedir.

2.2. Özellik Seçimi İle İlgili Literatür Çalışmaları

(Liu ve ark., 2011), karınca koloni optimizasyonu(KKO) ve kaba kümeleme algoritmasını bir arada kullanarak özellik seçimi metodunu geliştirmişlerdir. Geliştirilen bu yöntemle birlikte feromen güncelleme stratejisini daha da başarılı hale getirmeye çalışmışlardır.

(Manimala ve ark., 2011), hibrit yumuşak hesaplama tekniğini dokuz farklı enerji arızasını sınıflandırmak amacıyla özellik seçimi ve parametre optimizasyonu için önermişlerdir. Bu yaklaşımla ışıl işlem tabanlı yaklaşımların daha iyi sonuçlar elde ettiğini göstermişlerdir.

(Garcia ve ark., 2011), gen ifade mikro dizilerinde özellik seçimi metotları hesabı için kümelemeyle elde edilen indekslerin çok amaçlı optimizasyonu üzerinde çalışma gerçekleştirmişlerdir.

(Hacıbeyoglu, 2012), disjunktif normal formun, indirgenmiş fark fonksiyonundan ortaya çıkarılması yöntemini geliştirmiş, bu yöntemle karmaşıklık işlemi formun karekökünе kadar azaltmayı başarmıştır. Bu çalışma ile iki aşamalı lojik fonksiyon tabanlı yeni özellik seçimi yöntemi geliştirmiştir.

(de la Hoz ve ark., 2014), kendisini organize eden hiyerarşik haritalar yardımıyla ağ içindeki kusurların tespiti amaçlı yöntem geliştirmişlerdir. Geliştirilen bu yöntemin performans analizini görmek için DARPA/NSL-KDD veri kümesi kullanılmıştır.

(Ghamisi ve Benediktsson, 2014), salinas hiperspektral veri kümesi üzerinde yaptıkları çalışmada genetik algoritma ve parçacık sürü optimizasyonu ile birlikte kullanımıyla özellik seçimi gerçekleştirmişlerdir. Yöntemin uygun bir işlemci zamanı içerisinde en öğretici özellikleri otomatik olarak seçtiği gözlemlenmiştir.

(Olfati ve ark., 2014), göğüs kanseri teşhisinde destek vektör makineleri parametre optimizasyonu üzerinde yaptıkları çalışmada özellik azaltma için temel bileşen analizi (TBA), özellik seçimi için genetik algoritma ve sınıflandırma için de destek vektör makinalarını kullanmışlardır.

(Xue ve ark., 2014), arama aşamasında elde edilen daha önemli çözümleri depolayacak harici bir arşive sahip yeni bir PSO tabanlı özellik seçimi algoritması geliştirmişlerdir. Önerilen yöntemin PSOArR ve PSOArRWS isnminde iki özel metodu bulunmaktadır. 12 farklı benchmark fonksiyonu üzerinde yapılan deneysel çalışmalarda PSOArR ve PSOArRWS'nin tüm özellikler kullanılarak elde edilen başarıdan daha yüksek başarı elde ettikleri görülmüştür.

(Banka ve Dara, 2015), yüksek boyutlu özellik seçimi (ÖS) için, sınıflandırma ve validasyon yapmak amacıyla hamming uzaklık tabanlı ikili parçacık sürü optimizasyonu (İPSO) algoritmasını geliştirmişlerdir. Önerilen algoritmanın verimliliğini ve üstünlüğünü göstermek için üç farklı benchmark veri kümesi üzerinde deneysel çalışmalar detaylıca yapılmışlardır.

(Lin ve ark., 2015), yapay balık koloni algoritmasının lokal minimuma takılma ve çeşitlilik eksikliği gibi dezavantajlarından dolayı çalışmalarında 'modifiye edilmiş yapay balık koloni algoritması' nı (MYBKA) kullanmışlardır. MYBKA'ya dayalı destek vektör makinesi (DVM) için ÖS ve parametre optimizasyonu üzerinde çalışmışlardır. Bilinen UCI veri setleri üzerinde yapılan deneysel sonuçlarda daha az özellikli alt kümeler kullanarak sınıflandırma doğruluğu bakımından MYBKA'nın üstünlüğü göstermişlerdir.

(Moradi ve Rostami, 2015), sınıflandırma problemlerini çözmek için üç aşamadan oluşan graf kümeleme yaklaşımına ve (KKO) algoritmasına dayalı yeni bir ÖS yöntemi geliştirmişlerdir. Bu yaklaşımların ilkinde, tüm özellik kümesini bir graf olarak temsil etmişlerdir. İkinci aşamada, bir ağ belirleme algoritması kullanılarak özellikleri belli bazı gruplara bölmüşler ve son olarak da üçüncü evrede, nihai özellik alt kümesini seçmek için KKO algoritmasına dayalı yeni bir arama stratejisi geliştirmişlerdir.

2.3. Yapay Öğrenme Teknikleri İle İlgili Literatür Çalışmaları

Yapay öğrenme tekniklerinden olan Bayes ve Grey Wolf Optimization(GWO) algoritmaları hakkında hangi alanlarda ve ne zamandan başladığına dair araştırma yapılmıştır. Araştırmada BAYES ağlarının 1920'li yıllardan günümüze kadar farklı alanlarda kullanıldığı sonucuna ulaşılmıştır. GWO algoritmasının geçmişi ise 2010 yılına dayanmaktadır. Bu konu ile ilgili yapılan bazı literatür çalışmaları Çizelge 2.2. ve Çizelge 2.3.'de sunulmuştur.

Çizelge 2.2. Bayes ağları literatür taraması (Akcaoglu, 2012)

Yazar	**Yıl**	**Çalışma**
Wright	1921	Tarım alanında gelişme için belirsizlik ve olasılık içeren durumlarda ilk defa grafiksel gösterimi kullanmıştır.
Nadkarni ve Shenoy	1999	BA geliştirme üzerine çalışmışlar ve ürün geliştirme kararı üzerinde farklı değişkenlerin etkilerinin analizlerinde kullanmışlardır.
Winkler	2001	Sağlık sektörü, hastalık üzerinde ilaç etkisi analizi için BA kullanmıştır.
Ülengin ve diğ.	2005	Türkiye enflasyon oranlarını BA yöntemini kullanarak incelemişlerdir.
Poku	2005	Dış ticaret ve para piyasalarında operasyonel risk yönetimi için BA oluşturmuştur.
Krause	2006	Tarımda verimlilik artışı ve kapasite geliştirme için Bayes Ağlarından faydalanmıştır.
Yücebaş	2006	Tıbbi karar destek sistemlerinin oluşturulması için BA tabanlı bir algoritma geliştirmiştir.
Karatepe	2007	Sağlıkta karar verme için çok kriterli karar verme ile BA'yı birlikte kullanmıştır.
Oteniya	2008	Veri madenciliği ve yapay zekâya dayalı çözümlerde BA kullanmıştır.
Inman	2008	Su talebi yönetim stratejilerinin oluşturulması için Bayes Ağlarından faydalanmıştır.
Carr	2008	Askeri strateji zekâsının incelenmesinde Bayes Ağlarının kullanılabilirliğini araştırmıştır.
Kişioglu ve Topcu	2009	Telekomünikasyon sektöründe iptal analizi için BA temelli bir çalışma yapmışlardır.
Çınar ve Kayakutlu	2010	Enerji sektöründe senaryo analizi için Bayes Ağlarını kullanmışlardır.
Jones ve diğ.	2010	Üretim endüstrisinde bakım planlama için bayes ağlarını kullanmışlardır.
Menaught ve Chan	2010	Üretim sektöründe belirsizlik içeren durumlarda karar almada bayes ağları kullanımına yönelik bir çalışma yapmıştır.
Lakka ve diğ.	2011	Medya sektörü analizi için Bayes Ağlarını kullanmışlardır.
Lockamy	2011	Tedarikçi geliştirme ve kıyaslama için Bayes Ağları yöntemini kullanmıştır.

Bayes Ağları adına yapılan son yıllardaki çalışmalar incelendiğinde çeşitli çalışma alanlarında bu algoritmaların kullanılabilirliği gösterilmektedir.

(Nadkarni ve Shenoy, 2001), bayes ağları kullanarak ‘Bayes Causal Map’ olarak adlandırılan yeni bir geliştirme üzerine araştırmalar yapmışlardır. Bu araştırmalar ile olasılık tabanlı grafiksel bir uzay temsili oluşturmaya çalışmışlardır.

(Winkler, 2001), sağlık sektöründe ilaçların hastalık üzerinde etkisi için bayes ağlarını kullanmış ve istatistiksel olarak sağlık sektöründeki problemleri gidermeye çalışmıştır. Sorunları basitten zora değerlendirmiş ve basit sorunların üzerinde etkili sonuçlar ortaya koymaya çalışmıştır.

(Sahin ve ark., 2004), Türkiye enflasyon oranlarını bayes ağlarını kullanarak incelemişler, geleceğe yönelik enflasyon tahminleri üzerinde birkaç vaka çalışmaları yapmışlar ve test sonuçlarını çıkarmışlardır.

(Adusei-Poku ve ark., 2007), Hollanda'da dış ticaret ve para piyasalarında operasyonel risk yönetimini Bayes ağları ile oluşturmuştur. Finans sektöründeki operasyon risk kayıpları üzerinde mikro düzeyde meydana gelebilecek sorunların çözümü ve bankanın döviz ve para piyasası çözüm sürecine yönelik çalışmalar yapmıştır.

(Perez-Minana ve ark., 2012), tarımda verimlilik artışı için Bayes ağlarından faydalanmıştır. İngiliz tarım sektöründe sera gazı emisyon yönetimi için bayes ağlarından yararlanmıştır. Aynı yıl içinde yüce tıbbi karar destek sistemlerinin oluşturulması için Bayes ağları tabanlı bir algoritma oluşturmuştur.

(Warner ve ark., 1992) Bayes kuralına dayanan ilk tıbbi uygulama sistemlerinden birini gerçekleştirmişlerdir. Bayes kuralının teşhislere göre gerekli problemlere uygulanmasını teorik ilgiden çok bir gereklilik olarak tanımlarlar.

(Cowie ve ark., 2007), parçacık sürü optimizasyonu öğrenme yöntemi ile Bayes Ağlarını birleştirerek veri madenciliği ve yapay zekâya dayalı çözümlerde kullanmışlardır.

(Inman ve ark., 2011), su talebi yönetim stratejilerinin oluşturulması için Bayes ağlarından faydalanmıştır. Kullanıcı grupları ile çevre karar destek sistemleri üzerinde vaka çalışması yaparak Sofya da su talep yönetimi için Bayes Ağlarından yararlanarak çözümler üretmeye çalışmışlardır.

(Carr, 2008), askeri strateji zekâsının incelenmesinde Bayes ağlarının kullanılabilirliğini araştırmıştır. (Kisioglu ve Topcu, 2011), telekomünikasyon sektöründe iptal analizi için Bayes ağları temelli bir çalışma yapmışlardır.

(Cinar ve Kayakutlu, 2010), enerji sektöründe senaryo analizi için Bayes ağlarını kullanmışlardır. Enerji politikaları için oluşturulan senaryolar üzerinde bayes ağları uygulanarak araştırmacılara destek olacak modeller üzerine çalışmışlardır.

(Jones ve ark., 2010), üretim endüstrisinde bakım planlama için bayes ağlarını kullanmışlardır. Zaman analiz çalışması uygulamak için sorumlu parametreleri tespit

ederek bayes ağı modelleme ile sistem gecikme oranlarını tespit etmek için bir model üzerinde çalışmışlardır.

(Menaught ve Chan, 2010), üretim sektöründe belirsizlik içeren durumlarda karar almada Bayes ağlarını kullanmaya yönelik bir çalışma yapmışlardır.

(Lakka ve ark., 2011), medya sektörü analizi için Bayes ağlarını kullanmışlardır. Multimedya üzerinde anlamsal analiz çıkarmak için hem görsel hem de metinsel bilgilerin işlenerek bayes ağları ile bir model oluşturulması yönünde çalışmalarda bulunmuşlardır.

(Lockamy ve McCormack, 2012), tedarikçi geliştirme ve kıyaslama için bayes ağları yönetimini kullanmışlardır. Bayes ağların kullanarak tedarikçi risklerini kıyaslama için bir metodoloji öngörmüşlerdir. Tedarikçiye ait tüm parametreleri Bayes Ağ modelinden geçirerek tedarikçi risklerini ortaya çıkarmaya çalışmışlardır.

(Altuntas, 2011), "İstatistiksel Model Seçiminde Bayesci Yaklaşımlar ve Bayes Faktörü" isimli tez çalışması yapmıştır. Bu çalışmasında bayes modelini detaylarıyla incelenmiş, farklı modellerde uygulamaları ile göstermiştir.

(Orhan ve Adem, 2012), naive bayes yönteminde olasılık çarpanlarının etkileri üzerine bir çalışma yapmışlardır. Çalışmada, basit yapısı ve yüksek başarısıyla bilinen Naive Bayes (NB) yönteminde kullanılan olasılık çarpanlarının sınıflandırmaya etkisini araştırmışlar, sınıf olasılığı çarpanının sınıflandırmaya çoğu zaman yarar sağlasa da bazen zarar da verebildiği göstermişlerdir.

(Avcı, 2015; Avcı ve ark., 2013), meme kanseri verileri üzerine hormon reseptör survival olasılık karşılaştırılması ile ilgili bir çalışma yapmıştır. Meme kanseri verilerinin 'Bayesci Sağkalım Analizi' ile incelenmesi üzerine çalışmada bulunmuştur.

(Akcaoglu, 2012), değer akış haritalarında darboğazların giderilmesi için Bayes ağlarını kullanarak senaryo üretimi çalışması yapmış, ürettiği senaryolar ile çamaşır makinası fabrikasında sorunları çözümleyecek bir uygulama geliştirmiştir. Bayes metodunu kullanarak üretim verimliliğini ve üretim kapasitesini artıracak bir çalışma gerçekleştirmiştir.

(Akar ve Gundogdu, 2013), Bayes teorisinin su ürünlerinde kullanım olanakları ile ilgili bir çalışma yapmışlardır. Bayes ve istatistiksel yöntemler uygulanarak boy ağırlık, balıkçılık parametreleri ve güven aralıkları gibi parametreleri tahmin etmeye çalışmışlardır.

(Cinicioğlu ve ark., 2013), trafik kazaları analizi için Bayes ağları modeli kullanarak araştırmalar yapmışlardır. Trafik kazalarının nedenleri olan etmenleri Bayes

Ağlarını kullanarak oluşturdukları model aracılığıyla analiz etmişlerdir. Ortaya çıkan etmenlere göre ağın ve sonuçların yenilenebilmesi ve elde edilen sonuçların görsel bir şekil ile paylaşılması için grafiksel bir model üzerinde çalışmışlardır. Oluşturulan Bayes ağının doğruluk değerleri test verileri ile sınanmış geliştirilen modelin başarı sonuçları diğer modeller ile karşılaştırmışlardır.

Çizelge 2.3. GWO algoritması literatür taraması

Yazar	**Yıl**	**Çalışma**
L.I. Wang ve ark.	2010	GWO algoritması tabanlı algoritma ile Karmaşık Ekonomik Emisyon Dağıtım problemi üzerine araştırma yapılmıştır
C. Muro	2011	Av stratejileri için hesaplama simülasyonlarına üzerine çalışmalar yapmıştır.
L Korayem ve ark.	2013	GWO algoritması ile geliştirme yaparak araç yönlendirme problemleri çözümü yapmışlardır.
Feedforward Neural	2015	GWO algoritması ile çok katmanlı algılayıcılar için bir eğitim algoritması geliştirmiştir.
Emary et	2015	GWO ile özellik altküme seçim üzerine çalışmıştır.

GWO Algoritması için literatür taramasında ulaşılan kaynaklar şunlardır;

(Mirjalili ve ark., 2014) GWO üzerine detaylı bir çalışma yapmışlardır. Parçacık Sürü Optimizasyonu (Particle Swarm Optimization - PSO), Yerçekimi Arama Algoritması (Gravity Search Algorithm - GSA), Diferansiyel Gelişim Algoritması (Differential Evolution Algorithm - DEA) gibi birçok algoritmaya yakın sonuç verecek bir algoritma ortaya koymuşlardır. (Canis lupus) ismi verilen gri kurtların liderlik hiyerarşisi ve av mekanizmasını taklit ederek meta sezgisel bir yaklaşım ortaya çıkarmışlardır. Av ararken, av çevrelerken ve ava saldırırken gösterdikleri yaklaşımı modellemeye çalışmışlardır. 29 tanınmış test fonksiyonu ve test verileri üzerinde benchmarked testlerine tabi tutmuşlar ve PSO, GSA, DEA gibi diğer algoritmalarla rekabetçi sonuçlar ortaya koyduğu bilgisine ulaşmışlardır. Mühendislik tasarımı problemlerinde kullanıp geçerli sonuçlar elde etmişlerdir.

(Kamboj ve ark., 2016), GWO kullanarak dışbükey olmayan ekonomik yük dağıtım sorununun çözümü ile ilgili bir çalışma yapmışlardır. Bir meta-sezgisel arama algoritması olarak geliştirilen avı çevreleyen ve ava saldıran yaklaşım olarak benimsenen GWO algoritmasını elektrik güç sistemi olmayan dışbükey ve dinamik bir ekonomik yük dağıtım problemi (ELDP) çözümü için kullanmışlardır. Mukayese sonuçları GWO algoritması diğer iyi bilinen geleneksel, sezgisel ve meta-sezgisel arama algoritmaları ile

karşılaştırıldığında çok rekabetçi sonuçlar sağlamak için mümkün olduğunu göstermektedir şeklinde sonuca varmışlardır.

(Yusof ve Mustaffa, 2015), GWO algoritması kullanarak hammadde enerji Zaman serisi tahmini üzerine bir çalışma yapmışlardır. Yeni bir 'Swarm Intelligence' dayalı (SI) davranış, yani GWO ile kısa vadeli zaman serisi tahmin için geliştirilmiştir. West Texas Intermediate ham petrol ve benzin fiyatları üzerine model olarak diğer algoritmalarla karşılaştırılmıştır. Çalışma sonucunda GWO nun diğer sezgisel algoritmalara bir rakip olabileceği ortaya konulmaya çalışılmıştır.

(Muro ve ark., 2011), av stratejileri için hesaplama simülasyonlarına basit kurallar ortaya çıkarmıştır. Burada her kurt için ihtiyacı olan bilgi diğer kurdun konumu olarak değerlendirmiş ve uygun mesafe opsiyonunu bu bilgiye göre bulmaya çalışmışlardır. Av için minumum güvenli mesafe ve av doğru hareket zamanının tüm kurtlar arasında paylaşıldığının simülasyonu yapılmaya çalışılmıştır. Bu anlamda çeşitli sonuçlar ortaya koymuştur.

(Jayapriya ve Arock, 2015), birden fazla moleküler dizileri hizalanması için paralel GWO tekniği kullanarak bir model geliştirmeye çalışmışlardır. GWO algoritması çerçevesinde paralalde dizi sıralama çalışması yaparak ilk adımda moleküler dizileri hizalamaya çalışmışlardır. Sonuç olarak önerilen GWO algoritmasının diğer mevcut olanlara göre hesaplama süresini azaltır olduğunu göstermişlerdir.

(Wang ve ark., 2012), tarafından yapılan çalışmada gri tahminleme modeli kullanarak gıda güvenliği konusunda erken uyarı sağlayan bir sistem önerilmiştir. Gri tahmin modelleri ve süreç yeteneği indeksi kullanılarak bir değerlendirme ve erken uyarı tahmini yapmaya çalışmışlardır. Bir kalite indeksi inşa etmişler ve geliştirilmiş gri tahmin modeli sunmaya çalışmışlardır.

(Sulaiman ve ark., 2015), karma ekonomik emisyon dağıtım sorunlarını çözmek için GWO metodu üzerine bir uygulama geliştirmeye çalışmışlardır. Sistemdeki kombine ekonomik emisyon gönderme sorunu (CEED) çözmek için GWO metodu ile sonuca ulaşılmaya çalışılmıştır. GWO ile elde edilen sonuçları diğer optimizasyon teknikleri ile kıyaslamışlardır.

(Mirjalili, 2015), çok katmanlı algılayıcıların eğitiminde GWO algoritmasının ne kadar iyileştirici olduğunu göstermeye çalıştığı bir araştırma yapmıştır. Multi Layer Perceptron (MLP) ların eğitiminde GWO yu denemiş elde ettiği sonuçları PSO, ACO, GA, ES ve PBIL ile karşılaştırmıştır. GWO ile elde ettiği değerlerin çok rekabetçi olduğunu ve yakınlaştırmada yüksek değerler elde ettiğini sunmuştur.

(Mittal ve ark., 2016), global mühendislik optimizasyonu için GWO düzenlemesi yaparak sonuç çıkarmaya çalışmışlardır. Gerçek mekanik ve optik mühendisliği çözümü sorunları için GWO algoritmasını kullanarak keşif ve avlanma arasındaki uygun denge noktasını projelerde çözüm fikri olarak kullanmışlardır. Sonuçları test verilerine uygulamış ve başarı elde etmişlerdir.

(Korayem ve ark., 2015), GWO algoritması ile kapasite araç yönlendirme sorunu çözümü yapılmıştır. Araçların seyahatteki toplam maliyet ve mesafeyi minimize etmek için GWO algoritması kullanılarak bir çalışma yapılmıştır. Elde ettikleri sonuçları ve sorunları diğer algoritmalarla kıyaslamışlar GWO'nun bu problemin çözümünde yakın sonuçlar ortaya koyduğunu göstermeye çalışmışlardır.

(Aaghaee ve ark., 2014), çok katmanlı algılayıcı için GWO algoritması, bir eğitim algoritması olarak kullanılmışlardır. Optimizasyon problemini çözmek için geri beslemeli sinir ağlarının eğitiminde GWO algoritmasından yararlanmışlardır. Simülasyon da GWO'nun çok etkin sonuçlar çıkardığını göstermişlerdir.

(Wang ve Li, 2013), karmaşık ekonomik emisyon dağıtım problemine GWO algoritması tabanlı algoritma ile araştırma yapmışlardır.

(Niu ve ark., 2016), PM2.5 konsantrasyonu için CEEMD ve GWO melez yöntemini kullanarak bir grup model oluşturmaya çalışmışlardır. Önerilen karma ayrışma-topluluk modeli yüksek öngörü doğruluğu için tüm kabul kriter modellere son derece üstün sonuçlar çıkardığını ve yön tahmin oranlarının doğru olduğunu göstermeye çalışmışlardır.

(Shakarami ve Davoudkhani, 2016), zaman gecikmesi dikkate alarak GWO algoritmasına dayalı geniş alan güç sistemi dengeleyicisi (WAPSS) tasarımı için bir yöntem önermişlerdir. Bu model ile yaklaşım arası alan salınımları sönümlemenin yanı sıra uzaktan geri besleme sinyalleri haberleşmede gecikme yıkıcı etkileri telafisinde son derece etkili olduğunu göstermişlerdir.

(Lal ve ark., 2016), hidroelektrik terminallerindeki TCPS li Multi-Area Power System üzerinde bulanık PID kontrollerinin dağılımı üzerine çalışmışlardır. PSO ile karşılaştırarak başarılı sonuçlar ortaya çıkarmışlardır.

(Sharma ve Saikia, 2015), klasik kontrol tabanlı termik güç santrallerinde GWO algoritması kullanarak Multi Area ST'ların otomatik üretim kontrollerini yapmaya çalışmışlardır. Çalışmalarında GWO'nun, PID kontrolörün performansı ile STPP olmadan sistemde zaman çözümünde, zirve aşmayı ve salınımlar büyüklüğüne yerleşme bakımından diğerlerinden daha iyi olduğunu ortaya koymaya çalışmışlardır.

3. VERİ SETİ VE VERİ SETİ ÜZERİNDEKİ İŞLEMLER

Bu bölümde tez çalışmasında kullanılan veri setinin oluşturulması ve ön işlem yöntemleri hakkında genel bilgiler verilmiştir. Öncelikli olarak veri seti oluşturulması ve oluşturulan veri seti üzerinde veri ön işleme tekniklerinin uygulanmasında izlenecek adımlar hakkında bilgiler verilmiştir. Yapılan bu çalışmalarla doğru bir eğitim setinin oluşturulması hedeflenmektedir. Eğitim seti üzerindeki alanlar için normalizasyon işlemi, özellik seçimi ve yapay öğrenme algoritmaları hakkında detaylı araştırmalar ve elde edilen sonuçlar paylaşılmaya çalışılmıştır.

3.1. Veri Setinin Oluşturulması

Tez çalışmasındaki amaçlardan biri, müşteri hakkında yeterli verinin bulunmadığı durumlarda makine öğrenmesi yöntemleri yardımıyla kesinliği yüksek risk analizinin gerçekleştirilmesidir. Bu kapsamda yapılan çalışmalarda belli sayıda değer ve kayıt içeren özel bir kuruma ait veri seti kullanılmıştır.

Veri seti; sütunlarında niteliklerin, satırlarında ise o niteliklere ait değerlerin yer aldığı iki boyutlu martissel bir veri yığınıdır. Sütün ve satırların kesiştiği her alan hücre olarak isimlendirilir. Hücrelerin her birinde kesiştiği niteliğe ait olan sayısal ya da sembolsel ifadelerin gözlem değerleri tutulur. Buradaki değerler niteliğe ait gözlemlenen bilgilerin ifadesi olarak değerlendirilir. Hücrelerde herhangi bir değerin, sayısal ya da sembolik ifadenin olmaması o niteliğe ait gözlemin sonucunun olmadığı anlamına gelmektedir.

Veri setinin hazırlanması işlemi, çalışmalar ve gözlemler sonucunda elde edilen niteliklere ait verilerin iyi ölçüde toparlanmasından sonra geçerlidir. Elde edilen verilerin ve niteliklerin belli bir düzen içerisinde çizgisel ya da matris ifade ile gösterimi sağlanırsa veri seti ortaya çıkarılmış olur.

Veri setinde gözlemlenen sonucun olmadığı boşluklar kayıp veri olarak isimlendirilir. Çalışmalar ve gözlemler sonucu bulunan değerlerde kayıp veriler arttıkça, ölçülen niteliğin amaçlanan doğruluğa ulaşmasını sağlayacak veri sayısı azalır. Bu gibi durumlarda nitelikten herhangi bir sonuç çıkarımı yapılması oldukça zorlaşır. Bazı istatistiksel metotlarla belli orandaki kayıplar ihmal edilebilir ya da bazı yöntemlerle kayıp veri olan alanlara değer atama işlemi yapılabilir. Nitelikte kayıp değer oranı çok ise bu tür düzeltmeler gerçekçi olmayacağından sonuç olarak başarılı olmayabilir. Kayıp verinin çok olduğu durumlarda tekrar gözlem yapılarak sonuç bulunması gerekebilir.

Kayıp verinin çok olduğu durumda yeniden ölçüm ve gözlem yapmak atılacak en uygun adımdır. Çeşitli yöntemler kullanılarak eksik veriler düzenlenmelidir. Eksik verinin haricinde gürültülü veri de bir problem oluşturmaktadır. Gürültülü veri olması gerekenden farklı verilerin olduğu bilgi setidir. Bu verilerinde başarılı sonuçlara ulaşmak için veri seti içerisinden temizlenmesi gerekir.

Veri seti üzerinde niteliklerin iyi tanımlanmış olması çok önemlidir. Veri setinde ilgilenilen sonucu ortaya koyacak, temel özellik hakkında çıkarım yapacak, gerekli ve doğru niteliklerin belirlenmesi gerekmektedir. Elde edilen bu niteliklerin de en iyi bilgiyi verecek şekilde ölçeklendirilmesi doğru sonuçlara ulaşmak için önem ihtiva etmektedir.

3.2. Veri Madenciliği ve Veri Ön İşleme Teknikleri

Veri madenciliği kavramı eldeki verilerin değerlendirilerek anlamlı bilgilerin oluşturulmasıdır (MacKinnon ve Glick, 1999). Tıp, mühendislik, finans gibi birçok alanda başarı ile kullanılmaktadır. Özellikle tıp alanındaki birçok teşhis de veri madenciliği çalışmaları önemli oranlarda doğru sonuçlar elde etmiştir (Kusiak ve ark., 2000). Temel yöntem ham verideki gizli kalmış bilgileri ve ilişkileri tahmini bilgilere dönüştürmektir (Yan ve ark., 2001). Geliştirilen bu yöntem ve metotlar sayesinde veriler arası ilişkiler belirlenerek bu ilişkilere dayalı sonuçlar ortaya konulmaya çalışılmıştır. Bu ilişkilerin doğru kurulabilmesi için verilerin ön işlem tekniklerinden ve gerekli istatistiksel ve öğrenme algoritmalarından geçirilmiş olması gerekmektedir (Özkan M. ve Boran L. 2014).

Bilgisayar teknolojileri son yıllarda oldukça gelişim göstermiştir. Birçok alanda bilgisayar teknolojileri kullanarak gerekli analiz ve sistemsel çalışmalar yapılmaktadır. Özellikle makine öğrenmesi çalışmaları akıllı karar veren sistemler birçok alanda etkin bir şekilde kullanılmaktadır. Makine öğrenmesi çalışmalarının yapılabilmesi için en önemli konulardan bir tanesi çalışma yapılacak veri setinin sınıflandırma işlemidir. Veri sınıflandırma çalışmaları, veri madenciliği alanında en önde gelen konulardandır. Büyük miktarda veri içeren veri setleri üzerinden önemli olanları bulup çıkararak anlamlı bir veri alt kümesi elde etmeye veri madenciliği denir. Veri madenciliği büyük veri grupları içerisinden anlamlı özet veriler çıkarmayı hedefler.

Veri madenciliği çalışmalarında çalışmanın yapılacağı bir veri ambarının olması gerekmektedir. Veri ambarları konusu klasik veri tabanları üzerinden alınamayacak kadar çok verinin olduğu durumlarda değerlendirmeler yapabilmek için ilk kez 1991 yılında

William H. Immon tarafından ortaya atılmıştır (MacKinnon ve Glick, 1999). William H. Immon yaptığı çalışmada veri ambarını, zaman değişkeni kullanarak çeşitli kaynaklardan toplanan verileri yönetim kararlarını desteklemek amacı ile bilgi olarak sunulması şeklinde tanımlamaktadır. Özet olarak birçok veri tabanından alınan verilerin birleştirilerek depolanması işlemidir. Veri ambarlarının en önemli özelliklerinden bir tanesi kullanıcılara farklı katmanlarda detay sağlayabilmesidir. Detayın en alt katmanı arşivlenen kayıtlar ile ilgilidir. Daha üst katmanlar ise zaman bilgilerin daha fazla toplanmasıyla ilgilidir (MacKinnon ve Glick, 1999).

Veri madenciliğinin süreç olarak tanımlanması gerekirse; veri ambarı üzerinde yer alan çok fazla ve çeşitteki verinin çeşitli yöntemlerle analiz edilerek daha önce keşfedilmemiş bir bilginin/verinin ortaya çıkarılmaya çalışılması sürecidir denilebilir. Gizli kalmış verilerin değerlendirilerek karar verme mekanizmalarında kullanılması işlemidir. Bu tanımdan yararlanarak veri madenciliği çalışmalarının aynı zamanda bir istatistiksel çalışma süreci olduğunu da söylemek mümkündür (Statists, 1999).

Amerika Birleşik Devletleri'nde son 20 yıldır birçok alanda veri madenciliği algoritmalarının kullanıldığı bilinmektedir. Sahtecilik, vergi kaçakçılığı, gizli dinleme, suç tespiti gibi çeşitli alanlarda veri madenciliği algoritmaları kullanarak analizler yapılmıştır. Kaynaklar incelendiğinde sağlık sektöründe, tıp, biyoloji ve genetik gibi alanlarda veri madenciliği algoritmalarının çok sık kullanıldığı görülmektedir (Savas ve ark., 2012)

(Kusiak ve ark., 2000) yaptığı bir çalışmada, farklı zamanlarda farklı laboratuvarlardan toplanan test verileri üzerinde, veri madenciliği algoritmalarını uygulamışlar ve teşhiste %100 oranında doğruluk sağlamışlardır.

Veri madenciliği süreci veri ön işlem teknikleri aşağıdaki maddelerdeki gibi detaylandırıla bilinir (Cosku, 2013).

i. Verinin Temizlenmesi
ii. Verinin Bütünleştirilmesi
iii. Verinin İndirgemesi
iv. Verinin Dönüştürülmesi
v. Veri Madenciliği Algoritmalarının Uygulanması
vi. Sonuçlar ve Değerlendirmeler

3.2.1. Verinin Temizlenmesi

Veri seti içerisinde yer alan hatalı ve tutarsız verilere gürültülü veri denir. Veri setlerindeki gürültülü veriyi temizlemek için, eksik değerlerin olduğu alanlara sabit değerler atanabilir ya da diğer verilerin ortalaması alınarak eksik değerlerin bulunduğu alanlar ortalama değerler ile doldurulabilir. Bu işleme veri tamamlamada denilebilir. Veri temizlemek için bir diğer yöntem de eksik değer içeren kayıtlar veri setinden çıkarılarak veri atma işleminin uygulanmasıdır. Ayrıca verilere karar ağacı, regresyon gibi algoritmalar yardımıyla uygun bir tahmin yapılarak bulunan değer eksik olan kısımda kullanılabilir (Kaplan ve Gozen, 2010). Sonuç olarak eksik verilerin ya eğitim setinden çıkartılması ya da doğru sayılabilir değerlerle eksik verilerin tamamlanması işlemidir.

3.2.2. Verinin Bütünleştirilmesi

Farklı veri kaynaklarından ya da farklı veri setleri üzerinden elde edilen aynı bilgiyi taşıyan verilerin birlikte değerlendirilebilmesi için öncelikle tek tür veri yapısına dönüştürülmesi gerekmektedir. Örnek olarak cinsiyet veri tipi gösterebilir. Cinsiyet niteliği çok fazla veri tipinde tutulabilen bir niteliktir. Bazı veri setlerinde 0/1 şeklinde tamsayı veri tipinde tutulurken, bazı veri setlerinde K/E ya da Kadın / Erkek, bazı veri setlerinde de M / F ya da Male / Female şeklinde metinsel bir ifade ile tutulabilmektedir. Bu gibi durumlarda aynı tip bilgi taşıdıkları için tüm farklılıklar tek tip ve alanda bütünleştirilmesi gerekmektedir. Bu işleme veri bütünleştirme işlemi denilir.

Bilginin keşfinde ki başarı unsuru verilerin birbiriyle olan uyumlarına da bağlıdır. (Kaplan ve Gozen, 2010). Bu yüzden aynı anlamı ifade edecek veri tipleri aynı değerlerle birleştirilerek eğitim verisi üzerinde anlaşılabilir bir bütünlük sağlanarak çalışmaların daha doğru sonuçlar çıkarması hedeflenir.

3.2.3. Verinin İndirgemesi

Yapılacak çalışmalarda ihtiyaç olmayacakların veri seti içerisinden çıkartılması işlemidir. Veri madenciliği uygulamalarında bazı niteliklere ait bilgilerin araştırılan sonuca etkisi olmadığı düşünülüyorsa bu nitelikler ya da niteliğe ait bazı veriler veri setinden çıkartılarak veri seti boyutu azaltılabilir. Örneğin kişinin cep telefonu bilgisi sonuçlara etki etmiyorsa bu değişken veri setinden kaldırılabilir. Veri indirgeme

yöntemleri örnekleme, birleştirme, veri sıkıştırma, genelleme, boyut indirgeme olarak isimlendirebilir.

3.2.4. Verinin Dönüştürülmesi

Veri seti üzerinde işlem yapılacak verilerin, çalışmanın yapılacağı model ve uygulanacak algoritmalara göre veri setine ait niteliklerin içeriğini koruyacak şekilde başka değerlerle ifade edilmesidir. Burada uygulanacak model ve algoritma dönüşüm için oldukça önemlidir. Bu dönüşümün temel sebebi niteliklere ait ortalamalar ve varyans değerlerinin farklılığıdır. Niteliklere ait ortalama değerler ve varyansları birbirlerinden olukça farklı olduğu zaman büyük varyans ve ortalama değerlere sahip niteliklerin diğer niteliklere göre sonuç baskı değeri daha yüksek çıkar. Bu baskınlık bazı niteliklerin sonuca etkisinin az olması sonucunu doğurur. Bu gibi sebepleri önlemek için nitelikler arasında varyans ve ortalama değer farklarının açık şekilde olduğu durumlarda normalizasyon işlemi yapılmalıdır. Normalizasyon kısmında konu detaylı olarak ele alınmıştır.

3.2.5. Veri Madenciliği Algoritmalarının Uygulanması

Yapılan çeşitli işlemlerle araştırmalara hazır hale getirilmiş çalışma verileri veri seti olarak sisteme tanımlanır. Veri setini net bir şekilde oluşturulduktan sonra yapılacak çalışmaya göre uygun algoritmalar seçilir. Algoritmalar kullanılarak geçerli sonuçlar elde edilir ve sonuçlar düzenlenerek ilgili kişi ve birimlere sunulur. Hangi algoritma uygulanmışsa sonuçların o algoritmaya uygun olan çıktı gösterimi ile sunulması gerekmektedir. Örneğin çalışma modelinde hiyerarşik kümeleme yöntemi uygulanmışsa sonuçlarında ‘dendrogram grafiği’ olarak ilgililere sunulması uygundur.

3.2.6. Sonuçlar ve Değerlendirmeler

Veri setinin oluşturulmasından sonra yapılacak çalışma ile ilgili veri madenciliği algoritmaları uygulanır. Uygulanan algoritmalara göre ortaya farklı sonuçlar çıkabilmektedir. Burada sonuçların değerlendirilmesinde en önemli kıstaslardan birisi kullanılan algoritmaların çalışma yapılan alanlarda değerlendirilebilir sonuçların ortaya koyduğunun gerçekçiliğidir. Elde edilen sonuç ve sunumlar algoritmaların çalışma prensiplerine göre farklılıklar gösterebilmektedir.

3.3 Normalizasyon

Normalizasyon ayrık verilerin belli bir aralığa indirgemesi işlemidir. Verilerin iyi bir şekilde öğrenme algoritmalarına sokulması için sürekli ya da ayrık veri durumlarının göz önünde bulundurulması da gerekmektedir. Normalizasyon; veri setleri üzerinde sürekliliği değişmiş, veri kalitesinin ve kod yapısının bozulmuş olduğu durumlarda başvurulan bir işlemdir. Bu veri setlerine örnek olarak öğrenci notları, bilgi sistemlerinde tutulan kişilik verileri, maaş, tutar gibi mali verilerin yanı sıra insan kaynakları ve mali verilerin sistemde tutulmasını örnek gösterilebilir. Bu sebeple verinin normalleştirilmesi tekniklerinden bazıları aşağıdaki biçimde sıralanabilir (Roiger ve ark., 2003).

3.3.1. Ondalık Ölçekleme

Normalleştirmede en çok başvurulan yöntemdir. Ondalık Ölçekleme yönteminde, ilgili özellik değerlerinin ondalık kısımları değiştirilerek, artırılıp azaltılıp normalleştirme gerçekleştirilir. Bu ölçeklemeyle, sayısal verilerin -1 ile +1 arasında değer almalarını sağlayacak şekilde dönüştürülmesi işlemi amaçlanır (Wang ve Chen, 2007). Değiştirilen bu ondalık sayı değerleri, hareket eden özelliğin maksimum mutlak değerini ifade eder. Bu duruma örnek verilecek olursa *800* sayısı maksimum mutlak değeri belirtirse, basamak değeri *n=3* olacağından *800* değeri *0,8* olarak normalleştirilir (Oguzlar, 2003). İfade edilen işlem, aşağıda verilen denklem 3.1 ile hesaplanır. Bu eşitlikte (*i*) durum bilgisini, (*v*) özellik bilgisini *v(i)* ise özelliğin değer bilgisini, *k* sabit değer bilgisini ifade eder. Sonuçta [*0,1*] aralık değeri birim olarak ifade edilir.

$$v^1(i) = \frac{v(i)}{10^k} \qquad (3.1)$$

3.3.2. Min-Max Normalleştirme

Normalleştirmede kullanılan bir başka metot ise Min-Max normalleştirme yöntemidir. Bu yöntemle asıl veriler üzerinde lineer dönüşüm işlemi uygulanır. Yine bu metot ile datalar sıklıkla [*0,1*] aralığında bulunur. Min-Max yöntemi kapsamda ele alınan alan değerin, minimum değerden büyüklüğünü kıyaslar ve bu doğrultuda büyüklük farklarını sıralar (Wedding, 2005). İfade edilen işlem, aşağıda verilen denklem 3.2 yardımıyla hesaplanır. Burada x^* normalizasyon sonucu elde edilmiş veriyi, *x*

normalizasyon yapılacak girdi değerini, *max(x)* girdi setinde yer alan en büyük değeri, *min(x)* girdi setinde yer alan en küçük değeri ifade etmektedir. Sonuç [*min(x),max(x)*] aralığı birim cinsinden ifade edilir.

$$x^* = \frac{(x - min(X))}{aralık(x)} = \frac{(x - min(X))}{(max(X) - min(x))} \quad (3.2)$$

3.3.3. Z-Score Standartlaştırma

Normalizasyon işlemlerinde kullanılan bir başka metot ise Z-Score Standartlaştırma metodudur. Bu metotla istatistiksel veriler kullanılır. Eldeki verilerin ortalaması ile standart sapma sonuçları kullanılır. Çeşitli yöntemlerle yapılabilen normalleştirmeler, verilerin kapsamının ve boyutunun azaltılmasının yanı sıra, verilerle yapılabilecek işlemlerin daha küçük ve normalleştirilmiş veri kümesiyle etkin ve hızlı işlenip yorumlanması için de kullanılabilir (Khemka, 2003). Bu Eşitlikte x^* normalizasyon sonucu elde edilmiş veriyi, x normalizasyon yapılacak girdi değerini, standart sapma (x) girdi setinin standart sapmasını, ortalama(x) de girdi setinde yer alan tüm değerlerin ortalamasını ifade etmek için kullanılmaktadır. İfade edilen işlem, aşağıda verilen denklem 3.3 yardımıyla hesaplanır.

$$X^* = \frac{X - ortalama(X)}{standartsapma(x)} \quad (3.3)$$

3.4 Veri Düzeltme

Normalizasyon işlemlerinin uygulanarak aykırı verilerin temizlenmesi ve düzeltilmesi için iki algoritma üzerine çalışmalar yapılmıştır.

3.4.1. Veri Düzeltme İçin Veri Gruplama Metodu (Binning Methods)

Veri seti üzerindeki hatalı verilerin düzeltilmesi, tutarsız verilerin kaldırılması ve eksik verilerin tamamlanması işlemine veri temizleme denmektedir. Verinin işlenerek yapılacak çalışmalar için doğru sonuçların ortaya çıkarılabilmesi için veri seti üzerinde veri temizleme işlemleri oldukça önemlidir.

Gruplama metodundan yararlanarak veri güncellemek için üç metot önerilmektedir (Ozdemir, 2010). Bu yöntemler;

i. Ortalama Bularak Düzleştirme
ii. Ortancasını Bularak Düzleştirme
iii. Veri Sınırlarını Kullanarak Düzleştirme

Örnek veri seti: [*3, 8, 10, 15, 20, 22, 24, 25, 27, 29, 31, 33*] üzerinde;

1. Adım olarak yukarıdaki örnek veri seti eşit frekanslara ayrılarak dizi oluşturulacak olursa elde edilecek diziler Dizi 1, Dizi 2 ve Dizi 3 şeklinde ifade edilir.

- Dizi 1: *3, 8, 10, 15*
- Dizi 2: *20, 22, 24, 25*
- Dizi 3: *27, 29, 31, 33*

1.Adım uygulanarak eşit frekanslara ayrılarak elde edilen her bir dizi kendi içerisinde 'Ortalama Değerlere Göre Düzleştirme' işlemi uygulanacak olursa dizinin değer toplamları eleman sayısına bölünerek ortalama değerli bulunur. Dizideki her bir değer bulunan değerlere çekilerek 'Ortalama Değerlere Göre Düzleştirme' işlemi tamamlanmış olur. Ortalama değerlere getirilmiş durumu Dizi 1, Dizi 2 ve Dizi 3 şeklinde ifade edilir.

- Dizi 1: *9, 9, 9, 9*
- Dizi 2: *23, 23, 23, 23*
- Dizi 3: *30, 30, 30, 30*

1.Adım uygulanarak eşit frekanslara ayrılarak elde edilen her bir dizi kendi içerisinde 'Ortancasına Göre Düzleştirme' işlemi uygulanacak olursa dizinin her bir elemanının değerine dizinin ortanca elemanının değeri atanır. Dizi 1, Dizi 2 ve Dizi 3 şeklinde ifade ile dizilerin ortanca elemanlarının değerinin bütün dizi elemanlarına atanmış şekli sunulmuştur.

- Dizi 1: *8, 8, 8, 8*
- Dizi 2: *22, 22, 22, 22*
- Dizi 3: *29, 29, 29, 29*

1.Adım uygulanarak eşit frekanslara ayrılarak elde edilen her bir dizi kendi içerisinde ‘Sınırlara Göre Düzleştirme’ işlemi uygulanacak olursa dizi elemanına alt ya da üst sınır değerlerinden hangisine yakınsa değer olarak atanır. Dizilerin sınırlara göre değer atanmasının gösterimi Dizi 1, Dizi 2 ve Dizi 3 şeklinde ifade.

- Dizi 1: *3, 3, 15, 15*
- Dizi 2: *20, 20, 25, 25*
- Dizi 3: *27, 27, 33, 33*

3.4.2. Beş Sayı Özeti Metodu (The Five Number Summary Metod)

Normalizasyon yapmak aykırı değerleri temizlemek ve veri setindeki *min* ve *max* değerlerini elde etmek için Beş sayı özeti olarak bilinen beş sayı özeti metodu uygulanabilir. Bu metot veri setindeki değerler üzerinde aykırı değerlerin temizlenmesi ve veri setine ait *min* ve *max* değerlerinin elde edilerek bu değerlere göre normalizasyon işleminin yapılması için en çok kullanılan metotlardan birisidir.

Veri seti üzerinde beş sayı özeti metoduna ait tespit edilmesi gereken beş değer aşağıdaki değerlerdir.

i. Min-Minimum Değer
ii. Q1-First Quartile
iii. Med-Median
iv. Q3-Third Quartile
v. Max-Maximum

Bu metot ile yapılması gereken işlem veri seti üzerinden beş adet değer bularak bulunan değerlere göre veriden atılacak kısımların tespit edilmesi ve kalanlar için normalizasyon işleminin uygulanmasıdır. Bu değerleri elde etmek için veri seti küçükten büyüğe sıralanır. Sıralanan veriler eşit sayıda eleman içerecek şekilde 4 parça haline bölünür. Bölme işleminden sonra aşağıdaki hesaplar ile istenilen değerler bulunur. Q1 dörde bölünmüş ilk parçanın son sıra sayısını, Q3 te dörde bölünmüş üçüncü parçanın sor sıra sayısı değerlerini taşımaktadır. Bu değerlere bağlı olarak diğer değerler aşağıdaki denklemlerle hesaplanır.

$$Q1 = \frac{n}{4}\ ya\ da \frac{n+1}{4}$$

$$Q3 = \text{n} * \frac{3}{4} \text{ ya da } (\text{n} + 1) * \frac{3}{4}$$

$$\text{IQR} = \text{Q3} - \text{Q1}$$

$$\text{LF} = \text{Q1} - (1.5 * \text{IQR})$$

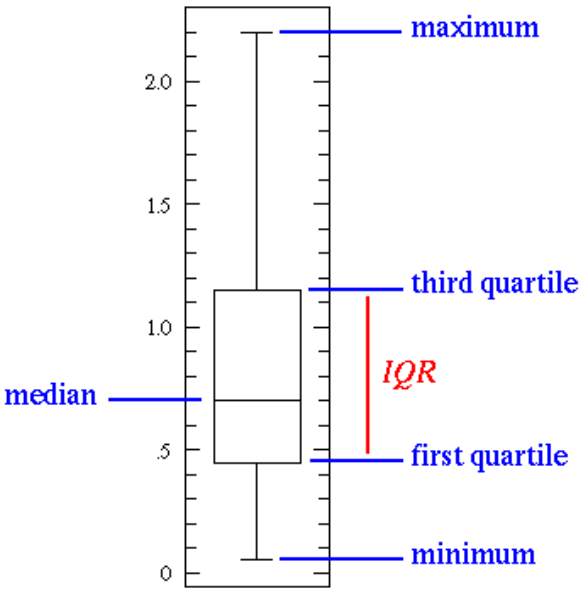
$$\text{UF} = \text{Q3} + (1.5 * \text{IQR})$$

$$\text{MIN} = \text{Listedeki} >= \text{LF deki ilk değer}$$

$$\text{MAX} = \text{Listedeki} \leq \text{UF ilk değer}$$

$$\text{Median} = \text{Sıralamadaki ortada yer alan değer}$$

Hesaplanan değerler veri seti üzerinde sahip olunan minimum, maximum ve ortalama değer gibi bilgileri bulunması sağlar. Bulunan min ve max değerine göre veri setinde yer alan aykırı değerler tespit edilir ve atılır. Aykırı değerler temizlendikten sonra var olan değerlerin min ve max bilgisine göre veri seti üzerinde normalizasyon çalışması yapılır. Şekil 3.1'de Beş Sayı Özeti metoduna ait Kutu Grafiği gösterimi sunulmuştur.



Şekil 3.1. Beş Sayı Özeti Kutu Grafiği Gösterimi (www.physics.csbsju.edu/stats/box2.html)

Tezde veri gruplama metodu (binning metod) kullanılarak *min* ve *max* değerleri elde edilmiş ve böylece aykırı değerlerin temizlenmesi ve aykırı değerleri atılmış niteliklerin normalizasyon işlemi yapılmıştır. Veri seti içerisinde yer alan finansal ve tutarsal veriler gibi sürekli verilerin [*0,1*] aralığında yayılması ve aykırı değerlerin tespit edilerek veri setinden atılması için bu metot kullanılmıştır.

4. OPTİMİZASYON ALGORİTMALARI

Bu bölümde öncelikle problemin çözüme ulaştırılması için kullanılacak özellik seçimi algoritmaları ve bu algoritmalardan ortaya çıkacak sonuçlara uygulanacak öğrenme algoritmaları hakkında bilgi verilmeye çalışılmıştır.

4.1 ÖZELLİK SEÇİMİ VE ÖZELLİK SEÇİMİ ALGORİTMALARI

Veri setinde oluşturulmuş özelliklere ait nicelik değerlerinin sonuca etkileri bir birine göre farklılıklar gösterir. Bir eğitim setindeki özellikler içinde yer alan değerlere göre sonuca etkisi yüksek olan, sonuca etkisi olmayan ilgisiz sütunların çıkartılması ve anlam gücü yüksek sütunlardan oluşan bir alt küme belirleme işlemine özellik seçimi denir. Genel olarak doğruluk ve ölçeklendirme için kullanılır. Çalışmada kullanılan özellik seçimi algoritmaları hakkında gerekli bilgi aşağıda sunulmuştur.

4.1.1. Bilgi Kazanımı (Information Gain) Algoritması

Bilgi Kazanımı (Information Gain), Entropinin (Dağınım, Dağıntı) tersi olarak tanımlanabilir. Bilgi kazanımı ifadesi düzensizliğini ifade etmek için kullanılır. Entropi ifadesini olasılık ifadesi olarak görebiliriz. Yani yazı tura atma işleminde gelebilecek sonuç %50 ihtimalli bir değerdir. Burada para atma işleminin adil bir şekilde yapıldığı ve sonuçların dengeli olacağı düşünülmektedir. Eğer her hangi bir hile söz konusu olacak olursa sistemin entropisi %50 den daha düşük olacaktır. Çünkü sistemde bir düzen yoktur. Hileli olan paranın tarafına doğru daha fazla sonuç gelecektir. Bu ifadeyi örnekleyecek olursak her atışta yazı gelen bir paranın ürettiği sonuçların entropisi 0'dır.

Entropi ifadesi ilk defa 'Shannon' tarafından veri iletişiminde ve bilgisayar bilimlerinde kullanılmıştır. Literatürde de Shannon Entropisi (Shannon's Entropy) olarak da geçen modele göre 'bir anlatımı ifade etmek için gereken en kısa ihtimallerin ortalama değeri alfabede bulunan tüm sembollerin logaritmasının elde edilen entropiye bölümüdür' (Seker, 2008) şeklinde bir kuram ortaya atmıştır. Yani ifade edilecek mesajdaki değişim ne kadar fazla olacak ise o kadar fazla kodlamaya ihtiyaç duyulmaktadır.

Bilgisayar bilimleri açısından daha kesin bir tanım yapmak gerekirse elimizdeki veriyi kaç bit ile (ikil) kodlayabileceğimize entropi ismi verilir. Örneğin bir yılda bulunan ayları kodlamak için kaç bit koda ihtiyacımız olduğunun gösterimidir.

Bilgi kazanımı entropinin tersidir ve [*0,1*] aralığında ifade edilir. Verilen bir niteliğin elde edilen sınıflandırma sonuçlarını ne kadarlık bir değer ile etkileyebileceğini gösterir. Her farklı sınıf için farklı farklı değerler alan bir nitelik varsa entropi *0* çıkacak ve bilgi kazanımı *1* olacaktır. Bu ifade seçilen nitelik ile sınıfın arasında birebir bir bağlantı olduğudur. Nitelik sınıfa ne kadar bağlıysa bilgi kazanımı o kadar yüksek, ne kadar bağımsızsa bilgi kazanımı o kadar düşük çıkar.

Bilgi kazanımı hesaplamaları yapılırken, veri setinde yer alana bütün niteliklerin ya da istenilen bir nitelik üzerinden gidilerek sonuç hesaplanabilir. Eğer veri seti içinde özellikle seçilmiş bir nitelik varsa bu niteliğe örnekleme (misal ya da sampling) adı verilir. Bütün veri seti üzerinden bu nitelik için hesaplama yapılır. Denklem 4.1 de gerekli değer hesaplama yöntemi sunulmuştur.

$$\sum_{i=1}^{k} \left(\left(\frac{frekans(Si,M)}{|M|} \right) . Log2(frekans(Si,M)/|M|) \right) \qquad (4.1)$$

Bilginin hesaplanması için kullanılması gereken formül denklem 4.1 de ki gösterim ile ifade edilebilir. Denklem 4.1 deki formül ile herhangi bir Misal (*M*) değeri için sınıfta (*S*) yer alan değerlerin frekansına bakılır. Denklem 4.1 de formülde |*M*| değeri, o sınıfta yer alan misallerin sayısını ifade etmektedir. Yapılacak çalışmalarda her örnek için bilgi (info) değeri hesaplandıktan sonra kazanım (gain) değerlerinin hesaplanması mümkündür. Bu hesaplama işlemleri sırasında bilgi belirli parçalara (partition) bölünür ve hesaplama işlemleri parçalar üzerinden yapılır. Parçalara ayırarak hesaplama işlemleri için gerekli yöntem denklem 4.2 de sunulmuştur.

$$Bilgix(P) = -\sum_{i=1}^{n} \left((\frac{|Pi|}{|P|} * Bilgi(Pi) \right) \qquad (4.2)$$

Her bir *i* verisi için bilgi hesaplamasını denklem 4.2 yardımı ile bulabiliriz. Bu hesaplama sayesinde kazanım değerini de bulabiliriz. Kazanım değeri ise denklem 4.3 de sunulan eşitlik ile hesaplanabilir.

$$\text{Kazanım(Özellik X)} = \text{Bilgi(P)} - \text{Bilgix(P)} \qquad (4.3)$$

Denklemler 4.1, 4.2 ve 4.3'e bakıldığında; istenilen bir *X* özelliğine ait kazanım değerini bulmak için, o niteliğin bağlı olduğu bütün parçaların bilgi hesabıyla o niteliğe ilgilendiren parçanın bilgi hesabının öncelikle bulunması gerektiği görülebilmektedir. Bu iki değer arasındaki fark niteliğe ait kazanım değerini verir.

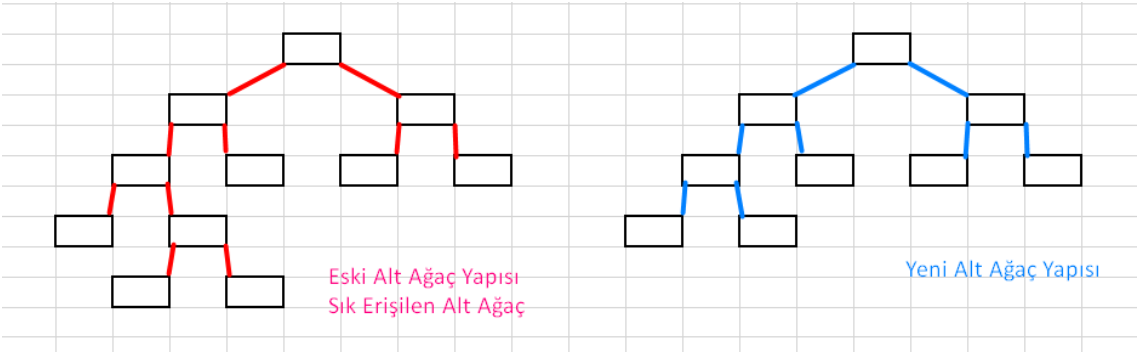
4.1.2. Kazanım Oranı (Gain Ratio) Algoritması

Kazanım Oranı algoritması Gain Ratio algoritması olarak bilinmektedir ve diğer bir anlamı olan C4.5 ağaçı olarak tanınmaktadır. Kazanım Oranı algoritması Bilgi Kazancı algoritmasının normalize edilmesi sonucu ortaya çıkmıştır. Bu yüzden Bilgi Kazancı algoritmasından gelişmiş olduğu söylenebilir. Bilgi kazanımı metodu çok çeşitli değerlere sahip nitelikleri seçme eğilimdedir. Böyle problemlerin çözümünde C4.5 kazan oranı kullanılır. Hesaplamalara ait denklemler denklem 4.4 ve denklem 4.5 de sunulmuştur.

$$SplitInfo\ A(D) = -\sum_{j=1}^{V} \left(\frac{|Dj|}{|D|} * Log2(\frac{|Dj|}{|D|})\right) \quad (4.4)$$

$$GainRatio(A) = Gain(A)/SplitInfo(A) \quad (4.5)$$

Kazanım Oranı karar ağaçlarına bir örnektir. C4.5 ağacında ve karar ağaçlarında hesaplanan entropi değerleri birer oran olarak tutulmaktadır. Algoritma gereği ağaç yapısı üzerinde, dallara erişim sıklıklarına göre alt ağaçların yer değiştirmesi ya da farklı seviyelere taşınması da mümkündür. Şekil 4.1.'de kazanım oranı ağaç yapısı sunulmuştur.



Şekil 4.1. Kazanım Oranı Ağaç Yapısı

Kazanım Oranı'nın çalışma prensibi aşağıdaki gibidir;

- Yapılan her işlemde adımda mevcut özellikler kontrol edilir.
- Her adım için özelliğin normalize edilmiş bilgi kazanım değerleri hesaplanır.
- Yeni karar düğümünün altına alt liste oluşturularak alt karar ağacı inşa edilir.

Veri kümesi için ilk olarak bilgi kazanımı hesaplanır. Bilgi kazanımı hesaplanırken, veri kümesindeki tüm nitelik ve hesaplanacak belirli bir veri üzerinden işlem yapılır. İşlem yapılacak olan bu veri grubuna örnekleme denilir. Elde edilen bu örneklem için bütün veri kümesi üzerinden hesaplama yapılır. Gerekli bilgi hesaplama yöntemi denklem 4.1 de sunulmuştur.

Bilgi kazanımı hesaplaması yukarıda anlatıldığı gibi yapılmaktadır. Denklem 4.1'deki formül ile herhangi bir Misal (*M*) değeri için sınıfta (*S*) yer alan değerlerin frekansına bakılır. Denklem 4.1'de formülde |*M*| değeri, o sınıfta yer alan misallerin sayısını ifade etmektedir. Yapılacak çalışmalarda her örnek için bilgi (info) değeri hesaplandıktan sonra kazanım (gain) değerlerinin hesaplanması mümkündür. Bu hesaplama işlemleri sırasında bilgi belirli parçalara (partition) bölünür ve hesaplama işlemleri parçalar üzerinden yapılır. Parçalara ayırarak hesaplama işlemleri için gerekli yöntem denklem 4.2'de sunulmuştur.

Kazanım bilgisi, birçok sonuçlanmış testlerde hesaplanmış ve bir ön yargı (ön bilgi) oluşturmuştur. Bu, en geniş değerlere sahip olan elementleri seçmeyi tercih etmesindendir. Örneğin benzersiz (unique) olarak tanımlanan bir öğe/özellik baz alınarak yapılan bir bölme (ayırma) işlemi, kaç kayıt varsa o sayıda dal verecektir. Çünkü her bölüm tektir. Bu bölme için veri setini sınıflandırmayı gerektiren bilgi *INFO(D)=0* olmalıdır. Bu nedenle, bu elemente göre yapılan bölümlendirme ile kazanım işlemine tabi tutulan bilgi en fazla olanıdır. Yani bölümlendirme, sınıflandırma için kullanışlıdır.

Sınıflandırma çalışmaları için izlenen bu yöntemler ön bilgilerin üstesinden gelmeye çalışan, kazanım oranı olarak bilinen, kazanım bilgisi uzantısını kullanır. En yüksek kazanım oranı değerine sahip özellik (element), ayırıcı (bölücü) özellik olarak seçilir.

Kazanım oranı algoritmasına, verilen veri seti içerindeki her bir niteliğin frekans cinsinden hesabını yaparak sonuca olan etkisini ortaya koyma işlemi olarak bakılabilir. Algoritma, entropi hesaplama yöntemi ile en etkili alanları hesaplayarak bir ağaç yapısı kurar.

4.2 SINIFLANDIRMA ALGORİTMALARI

Bu kısımda istatiksel sınıflandırma algoritmalardan olan BAYES algoritması ile sezgisel sınıflandırma algoritmalardan biri olan Gri Kurt Optimizasyon algoritması hakkında bilgi verilmektedir.

4.2.1 GRİ KURT OPTİMİZASYONU (GWO)

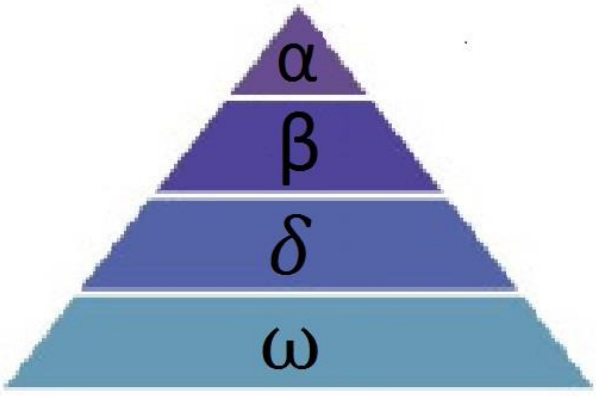
GWO algoritması sezgisel optimizasyon yöntemi olarak ele alınabilir. Sezgisel optimizasyon; Bir problemin çözümünü, doğruluğunun kanıtlanabilir olup olmadığını önemsenmeden fakat iyiye yakın olarak ele alma yöntemidir. GWO algoritması doğadaki gri kurtların liderlik hiyerarşisini ve avlanma mekanizmasını taklit eder.

Gri kurt hiyerarşisinde *Alfa*, *Beta*, *Delta* ve *Omega* olmak üzere 4 tip simülasyon uygulanır (Mirjalili ve ark., 2014).

Şekil 4.2.'de gösterilen gri kurt hiyerarşisinde lider dişi ya da erkek olabilir. Bu lider *Alfa* kurt olarak bilinir. *Alfa* kurt sürünün avlanmaya karar vermede, Uyku yeri, kalkma vakti vs. gibi olaylarda sorumluluk sahibidir. Hiyerarşide ikinci sırada bulunan kurt *Beta*'dır. *Beta* kurt karar verme ve diğer sürü aktivitelerinde *Alfa* kurdun yardımcısı olarak bilinir. *Beta* kurt, *Alfa* kurdun emirlerini diğer kurtlara iletir ve geri dönüş sağlar. Ayrıca *Alfa* kurdun sürü disiplinini uygulamada rol oynar. *Beta* kurt, *Alfa* kurdun yaşlanmasıyla ya da uzaklaşmasıyla onun yerine geçer.

Hiyerarşide en düşük seviyeli kurt *Omega*'dır. *Omega* kurt her zaman yemekte diğer kurtların doymasını ve ona sıra gelmesini bekler. *Omega* kurdu, her zaman baskın kurt seçer. *Omega* kurtlar önemsiz birey gibi görünse de kaybolması durumunda sürüde iç savaş ve problemler gözlenir (Mirjalili ve ark., 2014).

Hiyerarşide üçüncü sırada bulunan *Delta* kurdu ise eğer kurt *Alfa*, *Beta* ve ya *Omega* değilse *Delta* kurt olarak belirlenir. *Delta* kurdu, *Alfa* ve *Beta* kurtlar seçer fakat *Omega*'ya baskındırlar.



Şekil 4.2. Gri Kurt Hiyerarşisi (Mirjalili ve ark., 2014)

Ek olarak kurtların sosyal hiyerarşisinde grup olarak avlandığı ve bunu da sıraya koyduğu görülmektedir (Mirjalili ve ark., 2014).

- İzleme, takip ve ava yaklaşma
- Takip, çevreleyen ve durana kadar avını hareket ederek rahatsız etme
- Son olarak avına saldırma

Şekil 4.3.'de gösterilen avlanma hiyerarşisinde gri kurtlar öncelikle avlanma davranışına girerler(A), ardından sırasıyla takip, yaklaşma ve avını izlerler(B-C-D), son olarak takip, rahatsız etme ve çevreleme işlemini yaparlar(E).



Şekil 4.3. Gri Kurt Avlanması (Mirjalili ve ark., 2014)

GWO algoritması matematiksel modellemeye göre seviyelendirilir. Bu sosyal hiyerarşisinin alt bölümleri olan izleme, çevreleme ve ava saldırmadır.

4.2.1.1. Sosyal Hiyerarşi

Sırasıyla matematiksel modellemeyi yaparak GWO tasarımını sağlar. *Alfa* en iyi çözüm olarak kullanılır. Ardından sırasıyla *Beta* ve *Delta*, ikinci ve üçüncü çözüm olarak adlandırılır. Çözüme aday olarak da kalan *Omega* olarak kabul edilir. GWO algoritmasında av (optimizasyon) *Alfa*, *Beta* ve *Delta* tarafından yönlendirilir. *Omega* kurtlar ise bu üç kurdu izler (Mirjalili ve ark., 2014).

4.2.1.2. Avı Çevreleme

Yukarıda belirtildiği gibi, gri kurtlar av sırasında avı çevreler. Çevreleme davranışında sırasıyla matematiksel modellemede aşağıdaki denklem 4.6 ve denklem 4.7 gibidir (Mirjalili ve ark., 2014).

$$\vec{D} = | \vec{C} . \vec{X_p}(t) - \vec{X}(t)| \quad (4.6)$$

$$\vec{X}(t+1) = \vec{X_p}(t) - \vec{A} . \vec{D} \quad (4.7)$$

Burada *t* geçerli yinelemeyi gösterir.

A ve *C* vektör katsayısını,

Xp avın konumunun vektörü

X bir gri kurdun pozisyon vektörünü gösterir.

A ve *C* Vektörü aşağıdaki denklem 4.8 ve denklem 4.9 deki gibi hesaplanır:

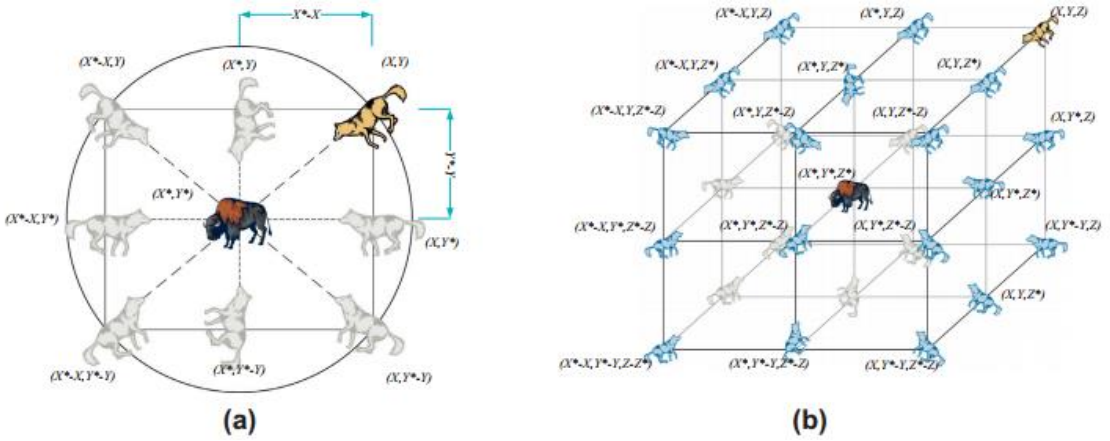
$$\vec{A} = 2\vec{a} . \vec{r_1} - \vec{a} \quad (4.8)$$

$$\vec{C} = 2 . \vec{r_2} \quad (4.9)$$

Burada $\tilde{a}$ bileşeni 2'den 0'a yineleme boyunca doğrusal azalmıştır ve *r1*,*r2* [*0,1*] arasında rastgele bir vektördür.

Şekil 4.4.'de görüldüğü gibi, iki boyutlu bir konum vektörü ve olası bazı komşular Şekil 4.4.(a)'da sunulmuştur. Bu şekilde görüldüğü gibi, (*x, y*) pozisyonunda gri kurt av pozisyonuna göre konumunu günceller (*X*, Y**). En iyi ajan *A* ve *C* vektör değerlerini güncel konumuna göre değiştirerek farklı yerlere ulaşabilir. Örneğin *A=(1,0)* ve *C=(1,1)* ayarına göre (*X*-X, Y**) değerine ulaşılabilir. 3D uzayda gri kurdun olası güncelleştirilmiş

pozisyonu Şekil. 4.4.(b)'de sunulmuştur. Yani bir gri kurt Şekil 4.4.'de kullanarak herhangi bir rastgele bir yerde av etrafı alanı içinde konumunu güncelleyebilir.



Şekil 4.4. 2D ve 3D pozisyon vektörleri ve bunların olası sonraki yerleri. (Mirjalili ve ark., 2014)

Aynı kavram *n* boyutlara sahip bir arama alanında uzatılabilir. Ve gri kurtlar şimdiye kadar elde edilen en iyi çözüm etrafında hiper küp (ya da hiper-küre) şeklinde hareket edecektir.

4.2.1.3. Avlanma

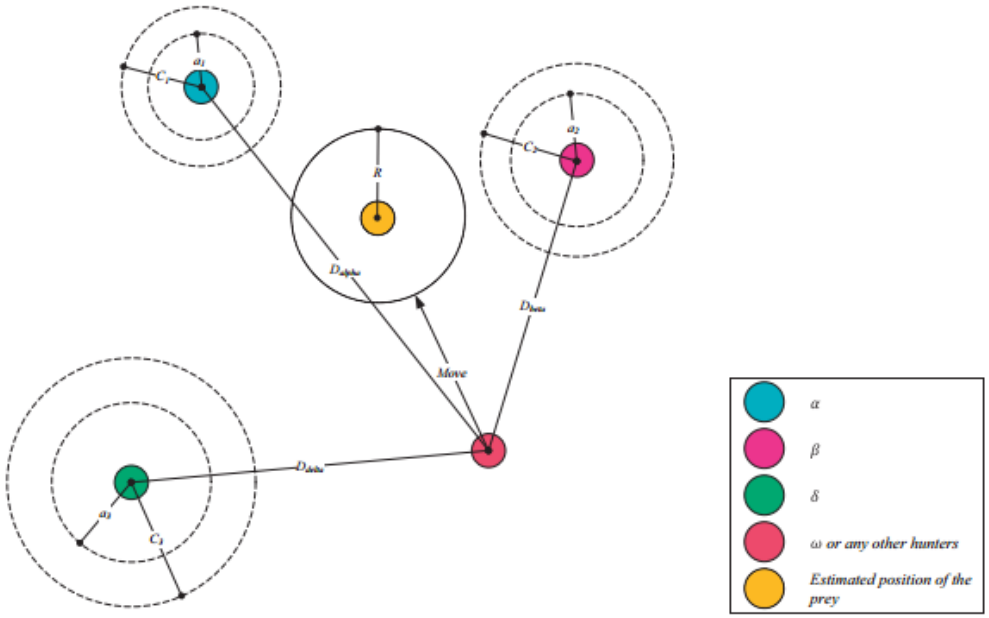
Gri kurt yerini tanıma ve orayı kuşatma yeteneğine sahiptir. Av genellikle *Alfa* tarafından yönlendirilir, *Beta* ve *Delta* da bazen ava katılabilir. Matematiksel olarak gri kurt av davranışını simüle etmek gerekirse, *Alfa* (en iyi aday çözüm) *Beta* olduğunu varsayalım ve *Delta* avın potansiyel konumu hakkında daha iyi bilgi sahibidir. Bu nedenle, şimdiye kadar elde edilen en iyi ilk üç sonucu kaydedilir ve en iyi arama ajanlarının konumuna göre diğer arama ajanlarının (*Omega*'lar dâhil) konumu güncellenir. Bu güncellemelere ait denklemler aşağıda denklem 4.10, denklem 4.11 ve denklem 4.12 ile gösterilmektedir (Mirjalili ve ark., 2014).

$$\overrightarrow{D_\alpha} = \left| \overrightarrow{C_1} . \overrightarrow{X_\alpha} - \vec{X} \right| \overrightarrow{D_\beta} = \left| \overrightarrow{C_2} . \overrightarrow{X_\beta} - \vec{X} \right| \overrightarrow{D_\delta} = \left| \overrightarrow{C_3} . \overrightarrow{X_\delta} - \vec{X} \right| \tag{4.10}$$

$$\overrightarrow{X_1} = \overrightarrow{X_\alpha} - \overrightarrow{A_1} . \left(\overrightarrow{D_\alpha}\right), \overrightarrow{X_2} = \overrightarrow{X_\beta} - \overrightarrow{A_2} . \left(\overrightarrow{D_\beta}\right), \overrightarrow{X_3} = \overrightarrow{X_\delta} - \overrightarrow{A_3} . \left(\overrightarrow{D_\delta}\right) \tag{4.11}$$

$$\vec{X}(t+1) = \frac{\overrightarrow{X_1} + \overrightarrow{X_2} + \overrightarrow{X_3}}{3} \tag{4.12}$$

Şekil 4.5.'de bir arama ajanı olan *Alfa*, *Beta* ve *Delta* 2D arama alanında uygun konumunu nasıl güncelleyeceğini gösterir. Son konum arama alanındaki *Alfa*, *Beta* ve *Delta* pozisyonlara göre tanımlanan bir daire içinde rasgele bir yerde olacağı görülmektedir. Başka bir deyişle *Alfa*, *Beta*, *Delta* avın konumunu tahmin eder ve diğer kurtlar av etrafında rastgele konumlarını günceller.

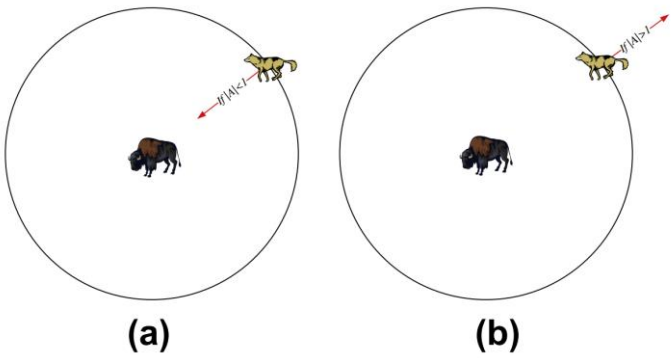


Şekil 4.5. GWO pozisyon güncelleme (Mirjalili ve ark., 2014)

4.2.1.4. Ava Saldırma (Sömürü)

Yukarıda belirtildiği gibi gri kurt av durduğunda ava saldırarak avı bitirir. Sırayla matematiksel modelle yaklaşan avın *a* vektörünün değerini azaltıyoruz. Böylelikle *a* vektörüne bağlı *A* vektörü de azalır. Başka bir deyişle *A* vektörü [*-2a, 2a*] değeri arasında rastgele bir sayıdır ve (*a*) yineleme boyunca 2'den 0 a düşürülmüştür.

A vektörü [*-1,1*] içinde rastgele değer olduğunda, arama ajanının bir sonraki pozisyonu geçerli pozisyonu ve avın pozisyonu arasında herhangi bir pozisyonda olabilir. Şekil 4.6.'da gösteriyor ki $|A|<1$ olduğunda kurt ava doğru saldırır (Mirjalili ve ark., 2014).

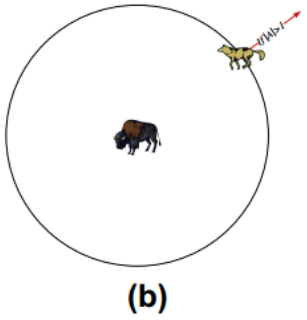


Şekil 4.6. Av Arama ve saldırma. (Mirjalili ve ark., 2014)

Şimdiye kadar önerilen operatörlerle, GWO algoritmasının arama ajanları olarak tanımlanan tüm ajanlar *Alfa*, *Beta* ve *Delta*'nın yerlerine göre konumlarını güncellerler ve ava doğru saldırır. Ancak, GWO algoritması bu operatörler ile yerel çözümlerde durgunluğa eğilimlidir. Önerilen çevreleme mekanizması keşfi bir dereceye kadar doğruyu gösterir. Fakat GWO keşif vurgulamak için daha fazla operatöre ihtiyaç duyar.

4.2.1.5. Av Arama (Keşif)

Gri kurtlar çoğunlukla *Alfa*, *Beta*, *Delta* ve *Omega* kurtların konumuna göre arama yaparlar. Gri kurtlar arama yapmak için dağılır ve avı bulduğunda saldırmak için toplanırlar. Şekil 4.7.'de gösterildiği üzere av durumu (uzaklık, yırtıcı olması vb.) gri kurdu zorlarsa gri kurt avdan sapacaktır (Mirjalili ve ark., 2014).



Şekil 4.7. Arama ve geri çekilme (Mirjalili ve ark., 2014)

Matematiksel olarak sapma modelinde, *1* den büyük rasgele değerler için *A* vektörü kullanılır. Tanımlanan *C* vektörü ise [*0,2*] rasgele değerlerini içerir. GWO yerel en yüksek değerden kaçınma durumunun lehine olarak, optimizasyon boyunca daha fazla rasgele davranışı göstermeyi sağlar. Kullanılan *C* vektörü, *A* vektörünün aksine

doğrusaldır ve azalmaya uğramadığını olmadığını belirtmek gerekir. Her zaman keşif esnasında rastgele değerleri sağlayabilmek için *C* gereklidir. Bu vektör bileşen, yerel optimal durgunluk durumunda çok yararlıdır. *C* vektörü ayrıca doğada av yaklaşırken ki engellerin etkisi olarak da kabul edilebilir. Genel olarak özetleyecek olursak, doğada engeller kurtların avlanma durumunda ortaya çıkar, kurdun hızlı ve rahat bir şekilde ava yaklaşmasını engeller, *C* vektörünün yaptığı da tam olarak budur. Özetle, av arama işlemi için GWO algoritması rastgele bir nüfus oluşturma işlemi ile başlar. Bu oluşturma işlemi esnasında yineleme, *Alfa, Beta, Delta* ve kurtların ve yırtıcının muhtemel konumunu tahmin edilir. Her aday çözüm yolu için av mesafesi güncellenir. Son olarak, GWO algoritması son bir kriter ile sonlandırılır. Bu algoritmanın sahte kodu Çizelge 4.1 de gösterilmektedir (Mirjalili ve ark., 2014).

Çizelge 4.1. GWO Algoritmasının sahte kodu (Mirjalili ve ark., 2014)

- Gri Kurt Popülasyonunu başlat
- Başlangıç değerlerini ata
- Herbir temsilci için uygunluk fonksiyon metodu hesapla
- X_α en iyi arama temsilcisini ata
- X_β en iyi 2. arama temsilcisini ata
- X_δ en iyi 3. arama temsilcisini ata
- TEKRARLA
    - HER BİR TEMSİLCİ İÇİ TEKRARLA
        - Temsilcinin posizyon bilgisini güncelle
    - Başangıç değerlerini güncelle
    - Herbir temsilci için uygunluk fonksiyonu metodu hesapla
    - $X_\alpha - X_\beta - X_\delta$ değerlerini güncelle
    - t = t+1
- Çevrim sayısı maksimum çevrim sayısından küçük olduğu sürece
- Geri çevir X_α

4.2.2 BAYES ALGORİTMASI

Bayes Teoremi; Thomas Bayes tarafından geliştirilen, koşullu olasılıkların hesaplanmasında kullanılan bir teoremdir. Bir olayın ortaya çıkmasında birden fazla bağımsız nedenin etkili olması durumunda, bu nedenlerden herhangi birinin o olayı meydana getirme olasılığını üzerine geliştirilmiştir. Bayes öğrenmesi, olasılıksal

çıkarıma bağlı yöntemler aracılığıyla, pratik yapıda öğrenme algoritmaları sunan bir yaklaşımdır. Bayes Teoremi temel anlamda, bir rassal değişken için olasılık dağılımı içinde koşullu olasılıklar ile marjinal olasılıklar arasındaki ilişkiyi göstermektedir. Bu yapısı ile bayes teoremi, bütün istatistikçiler için kabul edilir bir ilişkiyi açıklamaktadır (Pawlak, 2003).

Bayes öğrenmesi, üzerinde çalışılan her değerin, belirli olasılıksal dağılımlar içerisin de yer aldığını kabul etmektedir. Öğrenme süreci kapsamında en iyi kararın, ilgili eğitim örnekleri ile üzerinde çalışılan değerlerin olasılık dağılımlarının karşılaştırılması suretiyle elde edileceği düşünülmektedir. Bu öğrenme yaklaşımı ile olasılıksal tahminlerde bulunan hipotezler üzerinde çalışılabilmektedir.

Bu bağlamda, bir örneğin sınıf üyelik olasılığını kestirmektedir. Diğer bir deyişle, elde var olan, sınıflanmış durumdaki verileri kullanarak yeni bir verinin mevcut sınıflardan herhangi birine girme olasılığını hesaplayan bir yaklaşım olarak kullanılmaktadır. Bayes sınıflandırıcısı, istatistiksel sınıflandırma teknikleri arasında kabul edilmektedir. Bu sınıflandırıcı, en pratik öğrenme yaklaşımlarından birisidir. Yaklaşıma göre değerlerin hepsi aynı derecede öneme sahip ve birbirinden bağımsızdır. Genel anlamda sınıflandırma ve öğrenme problemlerinde sıklıkla kullanılmakta, özel anlamda tıbbi teşhis ve metin sınıflandırma gibi uygulamalarda oldukça başarılı olmaktadır (Karakoyun ve Hacıbeyoglu, 2014).

Bayes olasılık kuramı, matematiksel istatistik kuramının bir dalıdır (Siegmund-Schultze, 2004). Bu kuram; belirsizlik taşıyan herhangi bir durumun modelinin oluşturularak, bu durumla ilgili evrensel doğrular ve gerçekçi gözlemler ışığında belli sonuçlar üretilmesini sağlar.

Bağımlı olayların birbiri ile olan ilişkilerinin şartlı olasılık değerlerinin hesaplanmasını sağlayan bir teoremdir. Gözlemsel sonuçlardan yaralı bilgiler çıkarmak için tüm olasılıkların gözden geçirilmesi gereksinimi fikrinden dolayı ortaya çıkmıştır. Farklı sebeplerin aynı sonucu ortaya çıkardığı durumlarda, sonucu ortaya çıkaran sebebin ne olduğu bilinmeyebilir. Bu gibi durumlarda sonucu hangi sebebin ortaya çıkardığını tespit etmek için Bayes Teoremi kullanılmaktadır. Bir diğer deyişle bayes teoremi sonucu belliyken geriye doğru analiz yapmaktır.

İki olay arasındaki ilişkiyi, *A* olayına bağlı gerçekleşen *B* olayının (*A* olayı bilinirken *B* olayının incelenmesi) olma ihtimali, *B* olayına bağlı gerçekleşen *A* olayının (*B* olayı bilinirken *A* olayının incelenmesi) olma ihtimali değerleri birbirinden farklıdır.

Bu iki ters koşul arasında çok belirli bir ilişki vardır ve bu ilişkiyi ilk açıklayan istatistikçi Thomas Bayes olduğu için Bayes Teoremi denilmektedir.

Bayes ağları değişkenlere ait koşullu olasılık dağılımlarını ve değişkenlere ait alt kümeler arasındaki koşullu bağımsızlıkları tanımlamaktadır. Sınıflandırma amacıyla kullanılan istatistiksel algoritmalardır. Değişkenler arasındaki ilişki hakkındaki geçmiş bilgisini ne kadar iyi bilinirse o kadar iyi olasılıksal ilişki kurulur. Bayes ağları; bir dizi koşullu olasılıktan oluşan ve ilgili değişkenleri birbirine bağlayan Bayes Teoremi'ne dayanan grafiksel yapılardır. Bayes ağı olası durumları gösteren bir modeldir.

Genel anlamda, belirsizlik taşıyan bir sistemin Bayes Ağları'na göre modelini oluşturmak için şu adımlar takip edilmektedir. Sistemde önemli olan tüm değişkenler, düğümler şeklinde oluşturulur. Sistemdeki değişkenlerin birbirleriyle olan ilişkisi, düğümlerin bağlantısı şeklinde tanımlanmaktadır. Önceden bilinen ya da deneyimler sonucu elde edilmiş bilgiler ışığında koşullu olasılık değerleri belirlenir. İki düğüm arasındaki kenara "koşul kenar" denilmektedir. Bir koşul kenar, iki düğümü sebep-sonuç ilişkisi içinde birbirine bağlamaktadır.

Denklem 4.13'de Bayes teoremine ait formül verilmektedir. (Carlin ve Louis, 2008)

$$\mathrm{P}(A\backslash B) = \frac{\mathrm{P}\,(\mathrm{B}\backslash \mathrm{A})\,P(A)}{\mathrm{P}(\mathrm{B})} \qquad (4.13)$$

- *A* ve *B* rastgele olasılıklar olsun.
- *P(A)* : *A* olayının bağımsız olasılığı
- *P(B)* : *B* olayının bağımsız olasılığı
- *P(B| A)* : *A* olayının olduğu bilindiğinde *B* olayının olasılığı
- *P(A| B)* : *B* olayının olduğu bilindiğinde *A* olayının olasılığı

Avantajları;

- Geliştirilip uygulanması kolaydır.
- Genel anlamda "iyi sonuçlar" elde edilmesini sağlar.

Dezavantajları;

- Varsayıma dayalı bir yaklaşım söz konusudur.
- Nitelikler birbirinden bağımsız varsayılmaktadır ancak gerçek hayatta değerler birbirine bağımlı durumdadır.
- Değerler arası ilişkiler modellenememektedir.

4.2.2.1. Naive Bayes Sınıflandırma

Yalın bayes sınıflandırmanın ana fikri, bir belgenin sınıfının olasılığını tahmin etmek için verilen bir kelimenin sınıfının koşullu olasılıklarını kullanmaktır. Belge sınıflandırma gibi bazı öğrenme problemlerinde yaygın olarak kullanılan en pratik yaklaşımdır. Yalın Bayes'de Artımlı (Incremantal) olarak tabir edilen online bir öğrenme durumu vardır; her bir talim örneği artımlı olarak bir hipotezin doğru olma olasılığını arttırır veya azaltır. Öncül bilgi gözlemlenen verilerle birleştirilebilir. Varsayıma dayalı bir yaklaşımın söz konusu olması olumsuz bir yönüdür.

Bu sınıflandırıcı, en pratik öğrenme yaklaşımlarından birisidir. Geliştirilip uygulanması kolaydır. Değerlerin hepsi aynı derecede öneme sahip ve değerler birbirinden bağımsızdır. Naive Bayes sınıflandırması sadece sınıf dağılımlarını hesaplamakta böylece de hesaplama maliyetini azaltmaktadır. Genel olarak sınıflandırma ve öğrenme problemlerinde sıklıkla kullanılmakta ve iyi sonuçlar elde edilmektedir. Özel olarak da tıbbi teşhis ve metin sınıflandırma gibi uygulamalarda oldukça başarılı olmaktadır.

İstatistik olarak bağımsızlık önermesinden yararlanılarak BAYES teoreminde değer olarak verilmiş bir x değişkeninin $x = [\,x(1),\ x(2),\ \ldots\ ,\ x(L)\,]\,T \in RL$ formüle edilerek sınıf Si 'ye ait olup olmadığına karar vermek için kullanılan sınıflandırma yöntemine 'Naive Bayes' sınıflandırılması denir. Matematiksel bir ifadeyle gösterilmek istenirse denklem 4.14'de ki $P(x|Si)$ terimi yeniden denklem 4.15'de ki gibi yazılır. (Mitchell, 1997)

$$\mathrm{P}(x|Si)P(Si) > P(x\backslash Sj)P(Sj), \forall j \neq , i \tag{4.14}$$

$$\mathrm{P}(x|Si) \approx \prod_{k=1}^{L} P(xk|Si) \tag{4.15}$$

Böylece Bayes karar teoremi aşağıdaki şekli alır. Bayes karar teorisine göre $P(Si)\ ve\ P(Sj)$ i ve j sınıflarının öncel olasılıkları ise x sınıf Si 'ye aittir. Elde edilen bu veri kümesinde değerler denklem 4.16'daki gibi hesaplanabilir.

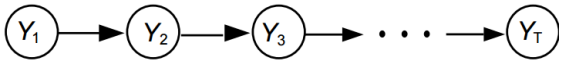
$$\mathrm{P}(Si) \prod_{k=1}^{L} P(\,xk\,|Si\,) > \mathrm{P}(Sj) \prod_{k=1}^{L} P\,(\,xk\,|\,Sj\,) \tag{4.16}$$

Naive bayes sınıflandırma yöntemi kullanarak yapılacak olan sınıflandırma çalışmalarının kullanım alanı her ne kadar az gözükse de, yüksek boyutlu dizilerde yeteri sayıda veriyle çalışarak *x* kümesi bileşenlerinin istatistik olarak bağımsız olması koşulu genişletilirse başarılı sonuçlar elde edilebilir.

4.2.2.2. Dynamic Bayes Ağı

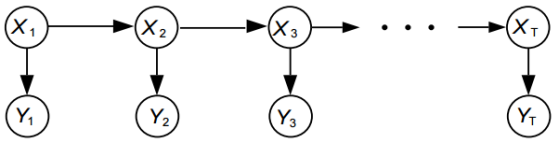
Dinamik bayes, naive bayes'den farklı olarak içinde 2 ayrı model içerir. Bunlar Hidden Markov ve State Space modelleridir. Zaman serisi modellemede, zaman içinde farklı noktalarda bazı değişkenlerin değerleri gözlemleniyor. Bir olay gelecekte başka bir olayı tetikleyebilir varsayılır (Murphy, 2002). Bayesian ağları yönlendirme okları zamanda ileri akmalıdır kuralı ile zaman serisi için tasarımı en basite düşürür. *t* index'inden her değişkene bir zaman atanır. (*Y1, … ,YT*) veri dizisi için en basit nedensel modellerden biri birinci dereceden Markov modelidir. Burada her değişken sadece doğrudan bir önceki değişken tarafından etkilenir. Şekil 4.8.'deki gibi gösterilir.

$$P(Y1, Y2, \quad \ldots, YT) = P(Y1)\, P(Y2|Y1) \quad \ldots \quad P(YT|YT-1)$$



Şekil 4.8. Birinci dereceden Markov işlemini temsil eden Bayes ağı.

Bu modeller doğrudan birden fazla adım üzerinden gözlemlenebilir niteliklerin arasındaki bağımlılıkları temsil etmemektedir. Gözlemlenebilen (*Y1, … , Y2*), modeli sadece *Yt*'den *Yt+1* değerine tahmin ederek kullanılabilir. Basit bir yolla, uzatılan Markov modelleri değişkenler arasında yüksek mertebeli etkileşime izin verir. Örneğin markov modeli T^{th}. Mertebeden gelen oklarda *Yt* için (*Yt-T, … , Yt-1*)'e izin verir. Başka yolla, geniş markov modeli gizli bir değişkene bağlı olarak gözlemlendiği varsayılırsa, bir markov işleyişinin durumlar dizileri Şekil 4.9.'da gösterildiği gibidir. Bu tür modeller klasik linear-gaussian durum-uzay modelidir ve kalman filtresi olarak bilinir (Huang, 2010).



Şekil 4.9. Bir durum-uzay modeli için şartlı bağımsızlık ilişkilerini belirten Bayes ağı

5. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

Bu bölümde tez boyunca yapılmış olan çalışmalar ve çalışmaların sonucuna ait bilgiler anlatılmıştır. Çalışma kapsamında;

- Veri setinin oluşturulması
- Oluşturulan veri setinin veri önişleme teknikleri kullanarak temizlenmesi
- Veri seti üzerinde özellik seçimi işleminin uygulanması
- Belirlenen özelliklere göre BAYES ve GWO metotlarının uygulanması
- Elde edilen sonuçların karşılaştırılması

Adımları uygulanmıştır.

Deneyler sonunda elde edilen sonuçların, performans değerlendirmeleri açısından anlaşılabilir bir biçimde yorumlanabilmesi için; performans karşılaştırma metriklerinin düzgün bir şekilde belirlenmesi ve tanımlanması gereklidir. Bu çalışmada aşağıdaki performans metrikleri kullanılmıştır:

- **Pozitif (P)**: **'Başarılı'** olarak doğru etiketlenmiş müşteri sayısı.
- **Negatif (N)**: **'Başarısız'** olarak doğru etiketlenmiş müşteri sayısı.
- **Doğru Pozitif (TP)**: **'Başarılı'** olarak doğru etiketlenmiş müşterilerin tahmin edilme sayısı.
- **Yanlış Negatif (FN): 'Başarısız'** olarak yanlış etiketlenen müşterilerin tahmin edilme sayısı. (Gerçekte 'Başarılı' olan müşterinin 'Başarısız' olarak etiketlenme durumu.)
- **Yanlış Pozitif (FP): 'Başarılı'** olarak yanlış etiketlenen müşterilerin tahmin edilme sayısı. (Gerçekte 'Başarısız' olan müşterinin 'Başarılı' olarak etiketlenme durumu.)
- **Doğru Negatif (TN): 'Başarısız'** olarak doğru etiketlenmiş müşterilerin tahmin edilme sayısı.

Burada "Başarılı" terimi "kredi verilebilirliği", "Başarısız" terimi ise "kredi verilemezliği" göstermektedir. Aslında problem iki sınıflı bir sınıflandırma problemidir. Performans matrikleri gerçek sonuç ve sınıflandırma sonucuna göre bir Matris biçiminde gösterilmektedir. Bu matris Karışıklık Matrisi (Class Confusion Matrix) olarak adlandırılmaktadır. Matrisdeki sütunlar tahmin sonuçlarına göre, satırlarsa gerçek

sonuçlara göre elde edilen sonuçların gösteriminde kullanılmaktadır. Bu değerler literatürde karşılaştırma ölçütü olarak kullanılacak değerlerin üretilmesinde kullanılmaktadır. Şekil 5.1.'de Karışıklık Matrisi gösterim şeması sunulmuştur.

		Öngörülen sınıf (Predicted Class)	
		C_1 (Positive)	C_2 (Negative)
Gerçek Sınıf (Actual Class)	C_1 (Positive)	True positive TP	False negative FN
	C_2 (Negative)	False positive FP	True negative TN

Şekil 5.1. Karışıklık Matrisi gösterimi

Tez kapsamında, ilgili performans metrikleri ile aşağıdaki karşılaştırma ölçütleri her bir test için hesaplanmıştır.

Accuracy-Doğruluk : (TP + TN) / (P + N) – Doğru Sonuçların Genel Popülasyona Oranı

Presisyon – Pozitif : TP / (TP + FP) – Kesinlik – Gerçek Pozitiflerin, tüm pozitiflere oranı

Presisyon – Negatif : TN / (TN + FN) – Kesinlik – Gerçek Negatiflerin, tüm negatiflere oranı

Type I Accuracy : FN / (P + N) – Doğruluk – Yanlış Negatiflerin Genel Popülasyona oranı

Bir müşteriyi gerçekte "Başarılı" olmasına rağmen "Başarısız" olarak etiketlemek "Type I Accuracy" terimi ile ifade edilir. 'Başarılı' olmasına rağmen 'Başarısız' olarak ifade edilen bir kaydın finansal açıdan yapılacak çalışmalarda daha kötü sonuçlar doğurabilme olasılığı bulunmaktadır. Bu sebeple tez kapsamında başarı oran kriteri olarak "Type I Accuracy" ölçütü tercih edilmiştir.

5.1. Çalışma Ortamı ve Uygulamanın Geliştirilmesi

Çalışmada bir finans kurumuna kredi talebinde bulunan 16088 müşteriye ait bilgiler kullanılmıştır. Müşteriye ait bilgilerden kullanılan özelliklerin isimleri ve veri tiplerinden bazıları Çizelge 5.4. ve Çizelge 5.6.'de sunulmuştur. Veri gizliliği ilkesi

nedeniyle tüm veri seti alanları verilmemiştir. 16088 kayıta ait 157 nitelik üzerinde çalışma gerçekleştirilmiştir.

Elde edilen 16088 kayıtlık veri kümesi MS SQL SERVER veri tabanı üzerinde tutulmak için gerekli alanlara uygun tablo tasarımı yapılmıştır. Tasarlanan veri tabanına ait tablolarda, veri setine ait bütün veriler depolanmıştır. Veri kaynaklarına erişecek ara yüzlerin geliştirilmesi yapılmıştır. Veri tabanında yer alan alanlar üzerinde veri ön işleme teknikleri kullanılarak aykırı veriler atılmış gürültüler temizlenmiş ve gerekli alanlara normalizasyon işlemleri uygulanmıştır. Sonuç olarak var olan veri seti veri ön işleme tekniklerinden geçirilip uygun hale getirilerek yanlış sonuçların ortaya çıkması engellenmeye çalışılmıştır.

Tez kapsamında yapılan çalışmada kullanılan veri seti farklı müşterilere ait bilgileri içermektedir. 16088 kayıttan oluşan veri kümesinde 13718 "Başarılı" sonuçlanmış, 2370 "Başarısız" sonuçlanmış kredi başvuru bilgisi içermektedir. Veri seti %16 olumsuz %84 olumlu sonuçlanmış başvurulardan oluşan 16088 müşteriye ait 157 nitelik taşıyan bilgilerden oluşmaktadır. Bu veri setlerini hazırlamak için var olan veri tabanı üzerinde önişlem tekniklerinden geçirilmiş verileri oluşturacak bir fonksiyon yazılmıştır.

Çalışmada Kazanım Oranı, Bilgi Kazanımı ve BAYES algoritmalarını kullanmak için WEKA uygulamasının C# dili için hazırlanmış olan açık kaynak kod alt yapısı kullanılmıştır. İnternet üzerinden geliştiricilerin kullanımına sunulmuş olan bu DLL ler sayesinde WEKA programında geliştirilmiş fonksiyonları Microsoft Visual Studio C# uygulaması içerisinden çağırmak mümkün hale gelmektedir. Bu işlem için kullanılacak veri kümesini WEKA standartlarına uygun ARFF dosyası haline getirmek gerekmektedir. Oluşturulan ARFF dosyaları ile de C# içinden WEKA fonksiyonları ile sonuç üretilebilmektedir.

Veri tabanında yer alan veri setine WEKA kütüphanesindeki algoritmalarda kullanabilmek için özellik seçimi algoritmalarında kullanmadan önce uygun formatta ARFF dosyası haline çevrilmiştir. Böylelikle geliştirme yapılan C# platformu içerisinde WEKA fonksiyonlarını kullanabilir hale gelmiş olmaktadır. Veri setlerindeki niteliklerin WEKA kütüphanesinde yer alan Kazanım Oranı ve Bilgi Kazanımı algoritmalarına göre özellik değerleri hesaplanarak elde edilmiş sonuçlar "5.2 Geliştirme Sonuçları" bölümünde sunulmuştur.

Veri seti için; Kazanım Oranı algoritmasının kullanılarak bulunan özellik ve özelliğe ait sonuçlar Çizelge 5.3.'de, Bilgi Kazanımı algoritması kullanılarak bulunan

özellik ve özelliğe ait sonuçları Çizelge 5.5.'de sunulmuştur. Bu çizelgelerde algoritmaların her bir nitelik için bulduğu değerler verilmiştir. Değeri 0 olan sütunlara çizelgede yer verilmemiştir. Burada özellik seçiminden elde edilen özelliklere ait değerler dikkate alınarak yapılan çalışmada değerleri yüksek ilk 16 sütun ve ilk 20 sütun alt küme olarak oluşturulmuştur. Bu alanlara ait verilerin WEKA üzerinde test işlemleri yapılmış ve 16 özellik üzerinde bulunan sonuçlar 20 özelliğe ait veriye göre bulunan sonuçlara çok yakın sonuçlar olduğu için performans ve zaman anlamında kazanım sağlamak için alt küme 16 özellikten oluşacak şekilde düzenlenmiştir.

Veri seti üzerinde yapılan çalışma sonucunda Kazanım Oranı ve Bilgi Kazanımı algoritmaları kullanılarak etki değerleri bulunmuş nitelikler etki değerlerine göre büyükten küçüğe göre sıralanmıştır. Kazanım Oranı algoritmasına göre sıralama sonucu yüksek değerli 16 nitelik Çizelge 5.4.'de, Bilgi Kazanımı algoritmasına göre sıralama sonucu yüksek değerli 16 nitelik Çizelge 5.6.'de sunulmuştur. Bu nitelikler kullanarak alt küme oluşturulmuş ve sınıflandırma çalışmaları oluşturulan alt küme ile yapılmıştır.

Elde edilen bu bilgilere göre gerekli ARFF dosyası oluşturulmuş, oluşturulan ARFF dosyası C# ile yazılan GWO algoritması ile çalıştırılarak elde edilen 'Başarılı' ve 'Başarısız' kayıt sayılarına ait sonuçın değerleri bulunmuştur. Gerekli GWO ya ait C# kodu Ekler kısmında 'EK-1 GWO Algoritmasının C# Kodu' olarak sunulmuştur. Aynı ARFF dosyası WEKA Kütüphanesi kullanılarak BAYES algoritması ile çalıştırılmış ve BAYES algoritmasından da elde edilen veriler kaydedilmiştir. Eğitim seti 5 parçaya bölünmüştür. K-Katlamalı Çapraz Doğrulama (K-Fold Cross Validation) yöntemi ile (Seker, 2013; Arlot ve Celisse, 2010) 4 parça (%80) eğitim verisi olarak, 1 parça (%20) ise test verisi olarak kullanılmıştır. K değeri 5 olarak seçilmiştir. BAYES ve GWO algoritması aşamasında veri setine ait bilgilerin %80 oranı kadar eğitim için %20 oranı kadarsa test için kullanılmıştır. Çapraz doğrulama ile her parça değiştirilerek 4 parça eğitim için bir parça test işlemi için kullanılmış elde edilen verilerin ortalaması alınarak sonuçlar bulunmuştur. Belli sayıda gerçekleştirilen iterasyonlarla elde edilen veriler Çizelge 5.1. ve Çizelge 5.2.'de sunulmuştur.

BAYES algoritması ile yapılan çalışmada istatistiksel bir sonuç elde edildiği için bir den fazla yapılan iterasyonda aynı sonuçlar bulunmuştur. Elde edilen sonuçlar aşağıdaki gibi Çizelge 5.1.'de sunulmuştur. Aynı veri seti %75 eğitim %25 test için K-Katlamalı Çapraz Doğrulama yapılarak da kullanılmıştır. %80 eğitim verisi sonuçları %75 eğitim verisi sonuçlarına yakın olduğu için GWO algoritması %80 eğitim verisi üzerinden sonuçlar çıkarılmaya çalışılmıştır. Verilere göre elde edilecek kazanım

değerleri sonuçlar kısmında gösterilmiştir. Bu sonuçların detaylı incelenmesi Çizelge 5.7.'de performans karşılaştırma çizelgesi olarak sunulmuştur.

Çizelge 5.1. BAYES Algoritması Sınıflandırma Değerleri

BAYES Algoritması Bulunan Sınıflandırma Değerleri									
Özellik Seçimi Algoritması	**Eğitim Seti Oranı**	**Test Seti Oranı**	**KS**	**P**	**N**	**TP**	**FN**	**TN**	**FP**
Kazanım Oranı	80%	20%	16088	13718	2370	12680	1038	1640	730
Kazanım Oranı	75%	25%	16088	13718	2370	12679	1039	1637	733
Bilgi Kazanımı	80%	20%	16088	13718	2370	12105	1613	1534	836
Bilgi Kazanımı	75%	25%	16088	13718	2370	12090	1628	1535	835

GWO algoritmasında bulunmak istenen temel amaç her bir özelliğin birbirine ve sonuca olan etkisi olduğu için öncelikle kolon değerleri sonuçlara göre sınıflandırılmıştır. Her bir özelliğin sonuca olan etkisi GWO algoritması yardımı ile hesaplanarak özellik etki değerleri bulunmuştur. Bu işlem yapılarak eğitim kümesinde sonucu "Başarılı" olarak kümelenen verilerin her bir özellik etki değeri hesaplanmış ve niteliklerin sonuca etki katsayısı çıkarılmıştır.

Aynı işlem eğitim setindeki sonucun "Başarısız" olduğu veriler içinde yapılmıştır. Test edilecek verilerin var olan değerleri sütun etki değerleri ile çarpılarak sonuca ulaşmaya hedefleyen bir uygunluk yöntemi metodu yazılmaya çalışılmıştır. Bu metot sonucunda bulunan değerlere göre "Başarılı" değer "Başarısız" değerden daha büyük bulunursa test verisinin sonucu "Başarılı", daha küçük bulunursa test verisinin sonucu "Başarısız" olarak işaretlenir. Bu işaretleme sonucu gerçek değeri ile bulunan değeri ayrı ayrı kaydedilir. Bu kaydedilen bilgilere göre P, N, TP, FN, FP, TN bilgileri çıkartılır. Bu bilgiler üzerinden Accuracy, Precision-P, Precision-N, Type I Accuracy değerleri hesaplanır.

GWO algoritması kullanarak elde edilen sonuçlar aşağıdaki gibi Çizelge 5.2.'de sunulmuştur. Bu veriler ile gerekli kazanım hesapları yapılmış elde edilen değerlere göre bulunan doğruluk sonuçları Çizelge 5.7.'de sonuç performans karşılaştırma çizelgesinde sunulmuştur.

Çizelge 5.2. GWO Algoritması Sınıflandırma Değerleri

GWO Algoritması Bulunan Sınıflandırma Değerleri									
Özellik Seçimi Algoritması	**Eğitim Seti Oranı**	**Test Seti Oranı**	**KS**	**P**	**N**	**TP**	**FN**	**TN**	**FP**
Kazanım Oranı	80%	20%	16088	13718	2370	12077	1641	1226	1144
Kazanım Oranı	75%	25%	16088	13718	2370	11797	1921	1331	1039
Bilgi Kazanımı	80%	20%	16088	13718	2370	11632	2086	1345	1025
Bilgi Kazanımı	75%	25%	16088	13718	2370	11425	2293	1308	1062

BAYES algoritması ile GWO algoritmasında elde edilen TP, FN, FP, TN bilgilerine göre doğruluk ve kesinlik değerleri hesaplanmıştır. Bulunan sonuç bilgilerine göre algoritmaların performansları karşılaştırılmaya çalışılmıştır. Kazanım değerleri üzerinden karşılaştırma yapılmış ve Kazanım Oranı algoritması ile belirlenen BAYES algoritmasının sonuçlarının Bilgi Kazanımı ile belirlenen BAYES algoritmasına göre daha başarılı olduğu gözlemlenmiştir. Ayrıca GWO algoritmasının da sınıflandırma işlemlerinde kullanılabilirliği ve sonuçların BAYES algoritmasına yakın olduğu tespit edilmiştir. Çizelge 5.7.'de sonuç performans karşılaştırma çizelgesi verilmiş ve başarı oranları sunulmuştur.

5.2. Sınıflandırma Sonuçları

Çalışmamız için bir finans kurumuna ait kredi talebinde bulunan 16.088 müşteriye ait 157 niteliğe sahip bilgi seti kullanılmıştır. Bu veri setinin WEKA Kütüphanesi kullanılarak Kazanım Oranı ve Bilgi Kazanımı algoritmalarında elde edilen sonuçlar aşağıdaki Çizelge 5.3. ve Çizelge 5.5.'de sunulmuştur.

Çizelge 5.3. Veri seti için niteliklerin Kazanım Oranı algoritması sonuçları

Nitelik Adı	**Değer**
AllotmentTypeId	0.236186958733662
TotBalanceOnlyMortgagesOpenAccountsLast3Months	0.22275348989911
CreditAmountNormalization	0.0409750216487406
PaybackStatusId	0.0331783288233053

MonthlyCommitmentForAllOpenAccountsCurrPymtStatus1_6	0.0247326671211851
TotBalanceInclMortgagesForAllOpenAccountsCurrPymtStatus1_6	0.0245479232958932
KKBScore	0.011573215096549
ProductTypeId	0.00815247943427406
ChildCount	0.007778724330567
TotBalanceExclMortgagesOpenAccounts12MonthsAgo	0.00714274911847424
TimeFromMostRecentClosedAccountsWorstPymtStatus3_6In12Months	0.00703108508242952
TimeFromMostRecentClosedAccountsWorstPymtStatus0_D_U_XIn12Months	0.00670851865010992
VehicleYear	0.0065701883592935
VechileStatusId	0.00656109529496808
TimeFromMostRecentDefaultForAllAccountsDefaultOrLegal	0.00605551111343483
TotBalanceExclMortgagesOpenAccountsLast3Months	0.0058121346082611
NumberOfOwnAccountsClosedAccountsWorstPymtStatus3_6In12Months	0.0054331355483316
NumberOfOwnAccountsClosedAccountsWorstPymtStatus0_D_U_XIn12MonthsNon	0.00537021891336487
MonthlyCommitmentMainJoint	0.00519199484010136
TotOutstBalAllRecsRetrieved	0.00505655094007421
TimeFromMostRecentClosedAccountsWorstPymtStatus0_D_U_XIn12MonthsNon	0.0047371887367637
TotBalanceExclMortgagesForAllOpenAccounts	0.00469678022677104
EstimateCostNormalization	0.00462196198701611
SGKTypeId	0.00455166382572905
NumberOfAccountsClosedAccountsWorstPymtStatus0_D_U_XIn12Months	0.00454767861113999
NumberOfAccountsClosedAccountsWorstPymtStatus3_6In12Months	0.00451119726379097
SpouseWorkStatusId	0.0044813045403519
JobId	0.00431317266027611
MontlyFamilyIncomeNormalization	0.00425954447113227
NumberOfOwnAccountsClosedAccountsWorstPymtStatus1_2In12Months	0.00418115930872783
NumberOfOwnAccountsClosedAccountsWorstPymtStatus1_2In12MonthsNon	0.004158669962374
EducationTypeId	0.00409802281795183
NumberOfOwnAccountsForAllAccountsDefaultOrLegalNon	0.00408518283104416
NumberOfAccountsForAllOpenAccountsCurrPymtStatus1_6	0.00398533052171818
ProductAdvancePaymentRateNormalization	0.00393058607307546
NumberOfOwnAccountsForAllAccountsDefaultOrLegal	0.00392903161487567
ApplicationReferenceNo	0.00391968032868789

VehicleMaturity	0.00391599299485792
CurrencyCode	0.00386816433573069
NumberOfAccountsForAllAccountsNotUpdated	0.00386816433573069
NumberOfAccountsClosedAccountsWorstPymtStatus0_D_U_XIn12MonthsNon	0.00386816433573069
SegmentVersionNumber	0.00386816433573069
NumberOfAccountsForAllAccountsNotUpdatedNon	0.00386816433573069
SegmentLength	0.00386816433573069
NumberOfAccountsClosedAccountsWorstPymtStatus1_2In12MonthsNon	0.00386816433573069
NumberOfAccountsForAllAccountsDefaultOrLegalNon	0.00386816433573069
NumberOfAccountsForAllOpenAccountsOtherCurrPymtStatus	0.00386816433573069
NumberOfAccountsClosedAccountsWorstPymtStatus3_6In12MonthsNon	0.00386816433573069
NumberOfOwnAccountsClosedAccountsWorstPymtStatus3_6In12MonthsNon	0.00386816433572566
CurrencyDivisor	0.00386816433572566
NumberOfAccountsClosedAccountsWorstPymtStatus1_2In12Months	0.00382708672318535
NumberOfOpenAccountsLast3Months	0.0038185754111513
SalaryNormalization	0.00378949255604891
SegmentSequence	0.00374661800576109
ReasonForApplicantDataReturn	0.00374661800575603
TotBalanceExclMortgagesOpenAccountsLast4_12Months	0.00370222237580549
NumberOfAccountsForAllAccountsDefaultOrLegal	0.00367723460901871
TotOutstBalAllRecsRetrievedNormalization	0.00332686781915663
SectorId	0.00328177441768373
HomeStatusId	0.0032216254828927
HasHome	0.0032216254828927
NumberOfAccountsOpenAccounts12MonthsAgo	0.00315757849089177
TimeFromMostRecentClosedAccountsWorstPymtStatus1_2In12Months	0.0031310289566314
TotalNoOfRecordsRetrieved	0.00307893897563688
NumberOfAccountsOpenAccountsLast4_12Months	0.00301087046218024
NumberOfOwnAccountsClosedAccountsWorstPymtStatus0_D_U_XIn12Months	0.00294510700049319
WorkTypeId	0.00283848205497019
SelfEmployeeIncomeNormalization	0.00273629604948489
NumberOfAccountsForAllOpenAccounts	0.00273397901038805
SpecialAccountStatusFlag	0.00269198579173952
NumberOfAccountsForAllOpenAccountsCurrPymtStatus0	0.00240979663179469
TotOutstBalanceForAllAccountsDefaultOrLegal	0.00237942665494556
MonthlyCommitmentMainJointNormalization	0.00196741113245031
HabitationPeriod	0.0011691170156184
WorkPeriod	0.000812749953575038
MaritalStatusId	0.000753888809460963

Çizelge 5.3.'de kullanılan veri setinde yer alan 157 nitelikten veri tipi numerik olan alanlara ait sütunların Kazanım Oranı algoritması yardımıyla hesaplanmış sonuç bilgileri gösterilmektedir. Burada veri seti üzerinde yer alan bütün alanların sonuca etkilerini bulabilmek için WEKA Kütüphanesi içerisinde yer alan Kazanım Oranı algoritmasından yararlanılmıştır. Algoritma her bir niteliğin sonuca etki değerini vermektedir. Bu değerler baz alınarak veri seti üzerinden bir alt küme oluşturulabilir. Elde edilen sonuçlara göre tez çalışmasında sonucu en yüksek çıkan 16 nitelik alınarak alt küme oluşturulmuş ve öğrenme fonksiyonları çalışmaları bu alt küme üzerinden gerçekleştirilmiştir. Çizelge 5.4. Kazanım Oranı Algoritması kullanılarak belirlenen 16 nitelik sunulmuştur.

Çizelge 5.4. Kazanım Oranı Algoritması kullanılarak belirlenen 16 nitelik

Attribute Name	**Nitelik Adı**
AllotmentTypeId	Finansman Türü
TotBalanceOnlyMortgagesOpenAccounts Last3Months	Son 3 ay Mortgate Borç Bakiye
CreditAmountNormalization	Kredi Tutar Bilgisi
PaybackStatusId	Geri Ödeme Durumu
MonthlyCommitmentForAllOpenAccount sCurrPymtStatus1_6	Tüm açık hesapların aylık ödeme yükümlülüğü
TotBalanceInclMortgagesForAllOpenAcc ountsCurrPymtStatus1_6	Tüm açık hesaplar üzerindeki toplam borç
KKBScore	KKB Skor Bilgisi
ProductTypeId	Ürün Tipi
ChildCount	Çocuk Sayısı
TotBalanceExclMortgagesOpenAccounts 12MonthsAgo	12 Ay önce açılan hesapların mortgate hariç borcu
TimeFromMostRecentClosedAccountsWo rstPymtStatus3_6In12Months	Son 12 ay içinde en yakın hesap kapanma tarihi
TimeFromMostRecentClosedAccountsWo rstPymtStatus0_D_U_XIn12Months	Tüm kapalı hesapların son 12 aydaki en kötü ödeme performansı
VehicleYear	Araç Yılı
VechileStatusId	Araç Durum Bilgisi
TimeFromMostRecentDefaultForAllAcco untsDefaultOrLegal	Hesapların en yakın kanuni takip ayı
TotBalanceExclMortgagesOpenAccounts Last3Months	Son 3 ay mortgage hariç toplam borç bakiye

Çizelge 5.5. Veri seti için niteliklerin Bilgi Kazanımı algoritması sonuçları

Nitelik Adı	Değer
CreditAmountNormalization	0.0511815799283305
KKBScore	0.0270633262662987
PaybackStatusId	0.0180663392558537
ProductTypeId	0.00970735470666306
TotOutstBalAllRecsRetrieved	0.00848737630893315
TotBalanceExclMortgagesForAllOpenAccounts	0.0082094087109249
MonthlyCommitmentMainJoint	0.00779047397021804
SGKTypeId	0.00751451448186846
TotBalanceExclMortgagesOpenAccounts12MonthsAgo	0.00700961389780108
NumberOfAccountsClosedAccountsWorstPymtStatus0_D_U_XIn12Months	0.00664322816714691
NumberOfAccountsOpenAccounts12MonthsAgo	0.00633433027230845
TimeFromMostRecentClosedAccountsWorstPymtStatus0_D_U_XIn12Months	0.00631821168538904
EstimateCostNormalization	0.00594545934842772
VehicleYear	0.00581675507049362
VechileStatusId	0.00580919388005752
VehicleMaturity	0.00577962589873804
TotalNoOfRecordsRetrieved	0.00541680284310686
SalaryNormalization	0.0052524989736753
ProductAdvancePaymentRateNormalization	0.00516867702246449
TotBalanceExclMortgagesOpenAccountsLast4_12Months	0.00484495028919618
NumberOfAccountsClosedAccountsWorstPymtStatus1_2In12Months	0.00473003212696166
NumberOfOpenAccountsLast3Months	0.00432255389459135
EducationTypeId	0.00429182269583694
NumberOfAccountsForAllOpenAccounts	0.00427886388036358
MontlyFamilyIncomeNormalization	0.00402028069179272
JobId	0.00396938051589213
NumberOfAccountsOpenAccountsLast4_12Months	0.00387789370901293
WorkTypeId	0.00386340602230428
NumberOfAccountsForAllAccountsDefaultOrLegal	0.00320532228039561
NumberOfAccountsClosedAccountsWorstPymtStatus3_6In12Months	0.00320088506948679
NumberOfOwnAccountsClosedAccountsWorstPymtStatus0_D_U_XIn12MonthsNon	0.00305358155398583
NumberOfOwnAccountsClosedAccountsWorstPymtStatus3_6In12Months	0.0029193371009486

NumberOfAccountsForAllOpenAccountsCurrPymtStatus1_6	0.00251358851762795
NumberOfAccountsForAllOpenAccountsCurrPymtStatus0	0.00248475642565726
NumberOfOwnAccountsClosedAccountsWorstPymtStatus0_D_U_XIn12Months	0.00238695683680323
HasHome	0.00236248318915477
HomeStatusId	0.00236248318915477
TimeFromMostRecentClosedAccountsWorstPymtStatus1_2In12Months	0.00235778568753364
AllotmentTypeId	0.00223590426675313
NumberOfOwnAccountsClosedAccountsWorstPymtStatus1_2In12Months	0.00222930839490965
NumberOfOwnAccountsClosedAccountsWorstPymtStatus1_2In12MonthsNon	0.00219879052866168
TotBalanceExclMortgagesOpenAccountsLast3Months	0.00217746052391843
NumberOfOwnAccountsForAllAccountsDefaultOrLegalNon	0.00212461309250966
NumberOfOwnAccountsForAllAccountsDefaultOrLegal	0.00203983552219666
ApplicationReferenceNo	0.00203924993012961
NumberOfOwnAccountsClosedAccountsWorstPymtStatus3_6In12MonthsNon	0.00200753212068183
CurrencyDivisor	0.00200753212068183
SegmentLength	0.00200753212068172
CurrencyCode	0.00200753212068172
SegmentVersionNumber	0.00200753212068172
NumberOfAccountsForAllAccountsNotUpdatedNon	0.00200753212068172
NumberOfAccountsForAllAccountsDefaultOrLegalNon	0.00200753212068172
NumberOfAccountsClosedAccountsWorstPymtStatus0_D_U_XIn12MonthsNon	0.00200753212068172
NumberOfAccountsClosedAccountsWorstPymtStatus3_6In12MonthsNon	0.00200753212068172
NumberOfAccountsClosedAccountsWorstPymtStatus1_2In12MonthsNon	0.00200753212068172
NumberOfAccountsForAllOpenAccountsOtherCurrPymtStatus	0.00200753212068172
NumberOfAccountsForAllAccountsNotUpdated	0.00200753212068172
SelfEmployeeIncomeNormalization	0.00196940728640693
SegmentSequence	0.00193216096484727
ReasonForApplicantDataReturn	0.00193216096484672
TotOutstBalAllRecsRetrievedNormalization	0.00191128640216953
MonthlyCommitmentMainJointNormalization	0.00181915669810662
TimeFromMostRecentDefaultForAllAccountsDefaultOrLegal	0.00158172087084496
TotBalanceOnlyMortgagesOpenAccountsLast3Months	0.00137529329961894

SectorId	0.0013183945741112
TimeFromMostRecentClosedAccountsWorstPymtStatus3_6In12Months	0.00117615343450594
HabitationPeriod	0.00116080361390691
SpouseWorkStatusId	0.001109276546156
TotBalanceInclMortgagesForAllOpenAccountsCurrPymtStatus1_6	0.00101547481500619
MonthlyCommitmentForAllOpenAccountsCurrPymtStatus1_6	0.000987014686781751
TotOutstBalanceForAllAccountsDefaultOrLegal	0.000870048105229282
WorkPeriod	0.000804397231345511
TimeFromMostRecentClosedAccountsWorstPymtStatus0_D_U_XIn12MonthsNon	0.000698336266449218
ChildCount	0.000688983240826113
SpecialAccountStatusFlag	0.000616910026024042
MaritalStatusId	0.000465088264826274

Çizelge 5.5.'de kullanılan veri setinde yer alan 157 nitelikten veri tipi numerik olan alanlara ait sütunların Bilgi Kazanımı algoritması yardımıyla hesaplanmış sonuç bilgileri gösterilmektedir. Burada veri seti içindeki bütün alanların sonuca etkilerini bulabilmek için WEKA Kütüphanesi içerisinde yer alan özellik seçimi algoritmalarından Bilgi Kazanımı algoritmasından yararlanılmıştır. Algoritma her bir niteliğin sonuca etki değerini vermektedir. Bu değerler baz alınarak Kazanım Oranı algoritmasında yapıldığı gibi veri seti üzerinden bir alt küme oluşturulabilir. Bilgi Kazanımı algoritmasının ortaya çıkardığı sonuçlarda Kazanım Oranı algoritmasında olduğu gibi sonucu en yüksek çıkan 16 nitelik alınarak alt küme oluşturulmuş ve öğrenme fonksiyonları çalışmaları bu alt küme üzerinden gerçekleştirilmiştir. Çizelge 5.6.'da Bilgi Kazanımı algoritması kullanılarak belirlenen 16 nitelik sunulmuştur.

Çizelge 5.6. Bilgi Kazanımı Algoritması kullanılarak belirlenen 16 nitelik

Attribute Name	**Nitelik Adı**
CreditAmountNormalization	Kredi Tutar Bilgisi
KKBScore	KKB Skor Bilgisi
PaybackStatusId	Geri Ödeme Durumu
ProductTypeId	Ürün Tipi
TotOutstBalAllRecsRetrieved	Toplam Borç Tutarı(Kendisi,kefil,ek hamil dahil)
TotBalanceExclMortgagesForAllOpenAccounts	Tüm açık hesapların mortgage hariç borcu
MonthlyCommitmentMainJoint	Aylık Ödeme Yükümlülüğü
SGKTypeId	SGK Tipi
TotBalanceExclMortgagesOpenAccounts12MonthsAgo	12 Ay önce açılan hesapların mortgate hariç borcu
NumberOfAccountsClosedAccountsWorstPymtStatus0_D_U_XIn12Months	12 Ay içinde tüm kapalı hesapların hesap sayısı
NumberOfAccountsOpenAccounts12MonthsAgo	12 Ay öncesi açılan hesap sayısı
TimeFromMostRecentClosedAccountsWorstPymtStatus0_D_U_XIn12Months	Tüm kapalı hesapların son 12 aydaki en kötü ödeme performansı
EstimateCostNormalization	Tahmini Maliyet Değeri
VehicleYear	Araç Yılı
VechileStatusId	Araç Durum Bilgisi
VehicleMaturity	Araç Vadesi

Çizelge 5.3. ve Çizelge 5.5.'de yorumlanması gereken en önemli husus var olan veri seti içerisinde sonuca etkisi yüksek olan niteliklerin ortaya çıkarılması için yapılacak olan çalışmadır. Tez çalışması kapsamında sonucu etkisi yüksek alt kümeyi belirlemede Kazanım Oranı ve Bilgi Kazanımı algoritmaları sonucu bulunan değerler baz alınmıştır. Burada niteliklerin sonuca etki değerlerini büyükten küçüğe sıraladığımız zaman en yüksek olan değerler seçilerek alt küme oluşturulması diğer işlemlerin sonuçlarının daha sağlıklı olmasını sağlayacaktır. Dikkat edilmesi gereken bir diğer husus ise niteliklerin bir biri ile olan ilişkisidir. Birbirine benzer iki nitelik sonucu aynı ölçüde etkileyeceği için bunlar arasındaki ilişkinin belirlenerek alt kümeye sadece birinin dâhil edilmesi gerekir. Örnekleyecek olursak Brüt Maaş ve Net Maaş verilerinden oluşan bir veri setinde iki nitelikte sonucu aynı yönde etkileyeceği için bunlardan birinin alt kümeye dâhil edilmesi yeterlidir. Nitelikler arası ilişkileri en güzel korelasyon analizi ile ortaya çıkarılabilir.

Özellik seçimi algoritmaları sonucu bulunan değerlere göre oluşturulan alt kümenin sınıflandırma algoritmalarında elde edilen kazanım sonuçları Çizelge 5.7.'de sunulmuştur.

Çizelge 5.7. Sınıflandırma Ölçütlerinin Performans Karşılaştırılması

	Veri Seti	**Performans Metrikleri**			
		Accuracy	**Precision - P**	**Precision - N**	**Type I Accuracy**
Kazanım Oranı + Bayes	16088 kayıtlı veri seti	0.8901	0.9455	0.6123	0.0645
Bilgi Kazanimi + Bayes	16088 kayıtlı veri seti	0.8477	0.9353	0.4874	0.1002
Kazanim Orani + GWO	16088 kayıtlı veri seti	0.8268	0.9134	0.4276	0.1020
Bilgi Kazanimi + GWO	16088 kayıtlı veri seti	0.8066	0.9190	0.3920	0.1296

Çizelge 5.7.'de doğruluk, pozitif ve negatif kesinlik ile yanlış bulunan verilerin veri setine oranını ifade eden 1.tip doğruluk bilgileri sunulmuştur. Bu bilgiler hesaplanarak Accuracy değeri yüksek olan algoritmaların başarısının yüksek olduğu söylenebilir. Accuracy-Doğruluk (TP + TN) / (P + N) , Presisyon-Pozitif TP / (TP + FP), Presisyon-Negatif TN / (TN + FN) ve Type I Accuracy FN / (P + N) bilgileri ile hesaplanmaktadır.

Bir müşteriyi gerçekte "Başarılı" olmasına rağmen "Başarısız" olarak etiketlemek "Type I Accuracy" terimi ile ifade edilir. "Başarılı" olmasına rağmen "Başarısız" olarak ifade edilen bir kaydın finansal açıdan yapılacak çalışmalarda daha kötü sonuçlar doğurabilme olasılığı bulunmaktadır. Bu sebeple tez kapsamında başarı oran kriteri olarak "Type I Accuracy" ölçütü tercih edilmiştir.

Bu anlamda çalışmada Type I Accuracy en düşük çıkan algoritmaların tercih edilmesi önerilmektedir. Type I Accuracy ile Accuarcy ters orantılı olduğu için Accuarcy değeri yüksek algoritmada seçilebilir. Tez çalışmasında algoritmalarla elde edilen sonuçlar birbirine yakın çıkmıştır. Bu sebeple Kazanım Oranı + Bayes veya Kazanım Oranı + GWO algoritmaları tercih edilerek yeni gelen bir kredi başvurusu yorumlanabilir.

6. SONUÇLAR VE ÖNERİLER

6.1 Sonuçlar

Çeşitli sektörlerde farklı alanlarda karşılaşılan sorunları çözmek için uygulanan yöntemlerden biri olan sınıflandırma problemini çözmek ve çözüm aşamasındaki performans ile başarı oranını artırmak için birçok yeni teknik ve algoritmalar denenmektedir. Bu tekniklerin ön adımı olan veri madenciliği aşamasında uygulanan veri seti oluşturma, veri ön işleme teknikleri, özellik seçimi ile veri sınıflandırma algoritmalarının araştırılması ve problemlerde uygulanması bu yüksek lisans tezinde ele alınmıştır. Veri seti olarak ele alınan eğitim verileri veri ön işleme tekniklerinden geçirilerek çalışmalara uygun bilgiler haline getirilmiştir.

Veri seti üzerinde yer alan niteliklerden alt küme oluşturmak için kullanılan özellik seçimi algoritmaları literatür de geniş bir kullanım alanına sahiptir. Birden fazla özellik seçimi algoritmaları mevcuttur. Kabul görmüş başarı oranı yüksek birçok algoritma vardır. Bu yüksek lisans tezinde Kazanım Oranı (Gain Ratio) ve Bilgi Kazanımı (Information Gain) algoritmaları ile çalışılmış ve özellik seçimi işlemi için bu algoritmalardan yararlanılmıştır. Kazanım Oranı ve Bilgi Kazanımı algoritmaları ile eğitim seti üzerinde özellik seçimi yapılmış elde edilen sonuçlar Bayes ve GWO algoritmaları verilerin sınıflandırılması için kullanılmıştır. Sınıflandırma algoritmalar farklı veri setleri üzerinde kullanılmış ve eğitim başarısı en yüksek olacak şekilde bir model oluşturulmaya çalışılmıştır. Elde edilen model ve veri setinde üzerinde birden fazla iterasyon denenmiş bulunan sonuçlara ait kazanım değerleri hesaplanarak gösterilmiştir. Bulunan sonuçlar başarı oranları üzerinden karşılaştırılmış ve sonuçları da sunulmuştur.

Özellik seçimi algoritmalarından Kazanım Oranı algoritması kullanarak modellenen sınıflandırmaların başarı yüzdelerinin Bilgi Kazanımı algoritmasına göre daha yüksek olduğu gözlemlenmiştir. Fakat hem Kazanım Oranı algoritmasında hem de Bilgi Kazanımı algoritmasında yüksek performans ve başarılı sonuçlar elde edilmiştir. Test başarılarının ise iterasyona ve popülasyon değerine göre değiştiği gözlemlenmiş olup genel olarak bakıldığı zaman test başarısının doğru, performans ve süre kıstaslarının iyi olduğu görülmüştür. Özellik seçimi algoritmaları kullanılarak alt küme oluşturma işlemi veri kümelerine ait özellik sayısı azaltılarak hesaplama maliyeti ve hesaplama süresini azaltmış bununla birlikte daha yüksek eğitim başarısı elde edilmiştir. Kazanım Oranı ve Bilgi Kazanımı algoritmalarının sınıflandırma çalışmalarında özellik seçimi algoritması olarak kullanılması başarılı sonuçlar ortaya çıkarmıştır.

Sınıflandırma, veri madenciliği ve makine öğrenmesi yaygın olarak kullanılan bir metodolojidir. Literatürde çok fazla sınıflandırma yöntemleri yer almaktadır. SVM RBF, SVM Polynomial, SVM Linear, Logistic Regrasyon, Naive Bayes, Dynamic Bayesian, Neural Network, Genetic Search ve SOM algoritmaları ile yapılmış birçok sınıflandırma çalışması mevuttur. Bu algoritmaları kullanarak birçok başarılı sınıflandırma çalışmaları yapılmıştır. Sınıflandırma probleminin çözümün de istatistiksel ve sezgisel olmak üzere farklı sınıflandırma yöntemi yaklaşımları mevcuttur.

Bu tezde istatistiksel sınıflandırmalardan biri olan BAYES ile sezgisel sınıflandırma yöntemlerinden biri olarak bilinen GWO algoritmaları kullanılmıştır. Var olan veri setinde yapılan iterasyon çalışmalarında elde edilen sonuçlar değerlendirilmiştir. Testlerde elde edilen modellere göre ortaya çıkan başarı sonuçları kıyas edildiğinde BAYES yönteminin de, GWO yönteminin de sınıflandırma çalışmalarında başarılı olduğu tespit edilmiştir. Özellikle sınıflandırma çalışmalarında elde edilen test sonuçlarına göre BAYES yönteminin yüksek başarı ve performansta çalıştığı gözlemlenmiştir. GWO algoritmasının sonuçların da başarı oranlarının BAYES e yakın olduğu fakat çalışma süresinin BAYES e göre daha yüksek olduğu, geliştirilebilecek metotlar sayesinde sınıflandırma işleminde kullanılabileceği tespit edilmiştir. GWO algoritmasının sınıflandırma işlemi gibi özellik seçimi aşamasında da kullanılabileceği sonucu ortaya çıkarılmıştır.

BAYES modeli sınıflandırma çalışmalarında kısıtlayıcı görülen bir önerme ile olasılık üzerine kurulmuş istatistiksel bir yaklaşımdır. Bu önermede sınıflandırma sonucuna etki edecek her bir niteliğin istatistiksel olarak bağımsız olması gerekmektedir. BAYES modeli, modeli yeniden yapılandırmaya gerek bırakmadan yeni eğitim veri setlerine kolayca uygulanabilirliği ve hızlı sonuç vermesi açısından kullanışlı bir modeldir. Fakat bunun yanında, veri setinde bulunan değişkenlerin birbirinden bağımsız olduğu aksiyomu bu yöntemin en büyük dezavantajlarından biridir.

Var olan eğitim verisi üzerinden hareket ederek sınıflandırma algoritmalarında elde edilen sonuç ve modele göre yeni bir kredi başvurusu değerlendirilebilir. Eğitilmiş sistemden ortaya çıkan öğrenme modeli dikkate alınarak yeni bir başvuru yapan kişinin bilgileri modele sunularak doğru sınıf bilgisine ulaşılabilir. Oluşturulan modele göre sonuç üretilir ve kredi verilebilirliği ya da verilemezliği tespit edilmiş olunur.

Tez çalışması sonuçlarına göre GWO ve BAYES algoritması kullanılarak sınıflandırma yapılabilir ve kredi başvuruları istatistiksel olarak bu algoritmalarla değerlendirilerek kredi verilebilirliği değerlendirilebilir.

6.2 Öneriler

Sınıflandırma algoritmalarında en önemli kısım eğitim aşaması ve doğru modeli ortaya koymaktır. Bu anlamda dikkat edilmesi gereken en öncelikli konu doğru bir veri seti üzerinde sistemi eğitmektir. Çalışmada ortaya konulan en önemli önerilerden birisi yapılacak sınıflandırma çalışmalarında veri setinin gerçekçi veriler içermesini sağlayacak gerekli ön işlemler uygulanmış doğru bir veri setinin kullanılmasıdır.

Uygun veri seti sağlandıktan sonra performansı artırmak için sonuca en çok etki gösterecek özelliklerin tespiti için özellik seçim yöntemleri uygulanabilir. Bu adımda başarısı kanıtlanmış algoritmalardan Kazanım Oranı, Bilgi Kazanımı, ID3 Karar Ağaçları, KN komşuluk gibi algoritmalar kullanılabilir. Yâda bu algoritmalardan ortaya çıkarılacak bir hibrit model üzerinden alt küme oluşturma (özellik seçimi) çalışması yapılabilir. Elde edilen modellerin sonuçları farklı hazır veri setleri ya da gerçek dünya problemlerinden elde edilen veri setleri üzerinde başarıları analiz edilebilir. Burada çok fazla sayıda özellik ve kayıt içeren veri setleri üzerinde çalışmak depolama ve hesaplama maliyetlerini artırmakta bununla beraber işlem sürelerini uzatmaktadır. Bunu önlemek için özellik seçimi algoritmaları yardımıyla bir alt küme üzerinde çalışmak hem hesaplama maliyetlerini azaltacak, hem işlem sürelerini kısaltacak hem de performans artıracaktır.

Veri alt kümesi oluşturulduktan sonrada çeşitli makine öğrenmesi yöntemlerine ait sınıflandırma algoritmaları kullanarak farklı modeller oluşturulabilir. Bu modeller üzerinde başarı değerleri hesaplanıp en uygun sınıflandırma yöntemi tespit edilebilir. Genelde modellerin farklı veri seti, farklı parametre değerlerinde birbirlerine üstünlüklerinin farklı olabileceği gözlemlenmektedir. Az sayıda nitelikten oluşan modellerde istatistiksel sınıflandırma algoritmaları, farklı bağlılık değerlerinden ve çok sayıda nitelik barındıran veri kümelerine ait modellerde ise sezgisel algoritmaların daha başarılı sonuçlar ortaya koyduğu yapılan çalışmalar ile gözlemlenmiştir. Bu tez çalışmasında GWO sınıflandırma modelinin BAYES sınıflandırma modeline yakın değerde sonuçlar verdiği, sınıflandırma problemlerinde ve özellik seçimi çalışmalarında GWO algoritmasında kullanılabilirliği önerilmektedir. Yalnız bu algoritmaların seçiminde niteliklerin bağımlılığı ve işlem süreleri dikkate alınması son derece önemlidir.

KAYNAKLAR

Aaghaee, N., Hayati, M. ve Valian, E., 2014, Feedforward neural network training using Grey Wolf Optimizer, *National Conference on Technology*, Energy and the Electrical and Computer Engineering Approach.

Abdou, H., Pointon, J. ve El-Masry, A., 2008, Neural nets versus conventional techniques in credit scoring in Egyptian banking, *Expert Systems with Applications*, 35 (3), 1275-1292.

Abdou, H. ve Pointon, J., 2011, Credit scoring, statistical techniques and evaluation criteria: A review of the literature, *Intelligent Systems in Accounting*, Finance and Management (18 , 2-3), 59-88.

Abdou, H. A., 2009, Genetic programming for credit scoring: The case of Egyptian public sector banks, *Expert Systems with Applications*, 36 (9), 11402-11417.

Adusei-Poku, K., Van den Brink, G. J. ve Zucchini, W., 2007, Implementing a Bayesian network for foreign exchange settlement: a case study in operational risk management, *Journal of Operational Risk*, 2 (2), 101-107.

Akar, M. ve Gundogdu, S., 2013, Bayes Teorisinin Su Ürünlerinde Kullanım Olanakları, *Journal of FisheriesSciences*, 8(1), 8-16.

Akcaoglu, O., 2012, Değer Akış Haritalarında Belirlenen Darboğazların Çözümü için Bayes Ağları ile Senaryo Üretimi: Çamaşır Makinası Fabrikasında Bir Uygulama, *Yüksek Lisans Tezi. T.C. İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü*, Endüstri Mühendisliği Anabilim Dalı, (Mühendislik Yönetimi Programı).

Akkoc, S., 2010, Yapay Sinir Ağları ve Doğrusal Ayırma Analizi ile Kredi Derecelendirme, *14. Ulusal Finans Sempozyumu*, 14, 157-171.

Altuntas, M., 2011, İstatistiksel Model Seçiminde Bayesci Yaklaşımlar ve Bayes Faktörü, *Yüksek Lisans Tezi. T.C. İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü*, T.C. Sinop Üniversitesi Fen Bilimleri Enstitüsü.

Avcı, E., Bekiroğlu, N. ve Yay, M., 2013, Meme Kanseri Verilerinin Bayesci Sağkalım Analizi İle İncelenmesi, *Uluslararası 8. İstatistik Kongresi*, Kemer-Antalya.

Avcı, E., 2015, Bayesian Survival Analysis: Comparison of Survival Probability of Hormone Receptor Status for Breast Cancer Data, *International Journal of Data Analysis Techniques and Strategies* (1559664).

Baesens, B., Setiono, R., Mues, C. ve Vanthienen, J., 2003, Using neural network rule extraction and decision tables for credit-risk evaluation, *Management Science*, 49 (3), 312-329.

Baklouti, I., 2013, Determinants of Microcredit Repayment: The Case of Tunisian Microfinance Bank, *African Development Review-Revue Africaine De Developpement*, 25 (3), 370-382.

Banka, H. ve Dara, S., 2015, A Hamming distance based binary particle swarm optimization (HDBPSO) algorithm for high dimensional feature selection, classification and validation, *Pattern Recognition Letters*, 52, 94-100.

Bao, Y. K., Hu, Z. Y. ve Xiong, T., 2013, A PSO and pattern search based memetic algorithm for SVMs parameters optimization, *Neurocomputing*, 117, 98-106.

Bekhet, H. A. ve Eletter, S. F. K., 2014, Credit risk assessment model for Jordanian commercial banks: Neural scoring approach *Review of Development Finance*, 4 (1), 20-28.

Bhaduri, A., 2009, Credit Scoring using Artificial Immune System Algorithms: A Comparative Study, *2009 World Congress on Nature & Biologically Inspired Computing (Nabic 2009)*, 1539-1542.

Blanco, A., Pino-Mejias, R., Lara, J. ve Rayo, S., 2013, Credit scoring models for the microfinance industry using neural networks: Evidence from Peru, *Expert Systems with Applications*, 40 (1), 356-364.

Carlin, B. P. ve Louis, T. A., 2008, Bayesian Methods for Data Analysis, *Texts in Statistical Science CRC Press*, 13:978, 15-35.

Chang, C. C. ve Lin, C. J., 2011, LIBSVM: A Library for Support Vector Machines, *Acm Transactions on Intelligent Systems and Technology*, 2 (3).

Chen, M. C. ve Huang, S. H., 2003, Credit scoring and rejected instances reassigning through evolutionary computation techniques, *Expert Systems with Applications*, 24 (4), 433-441.

Chen, S. Y. ve Liu, X. H., 2004, The contribution of data mining to information science, *Journal of Information Science*, 30 (6), 550-558.

Cinar, D. ve Kayakutlu, G., 2010, Scenario analysis using Bayesian networks: A case study in energy sector, *Knowledge-Based Systems*, 23 (3), 267-276.

Cinicioğlu, E., Atalay, M. ve Yorulmaz, H., 2013, Trafik Kazaları Analizi için Bayes Ağları Modeli, *Bilişim Teknoloji Dergisi*, 6 (2).

Cole, R. A., 1998, The importance of relationships to the availability of credit, *Journal of Banking & Finance*, 22 (6-8), 959-977.

Cosku, E., 2013, Veri Madenciliği,, *http://ab.org.tr/ab13/bildiri/175.pdf*.

Cowie, J., Oteniya, L. ve Coles, R., 2007, Particle Swarm Optimisation for learning Bayesian Networks, *World Congress on Engineering 2007, Vols 1 and 2*, 71-76.

Crone, S. F. ve Finlay, S., 2012, Instance sampling in credit scoring: An empirical study of sample size and balancing, *International Journal of Forecasting*, 28 (1), 224-238.

de la Hoz, E., de la Hoz, E., Ortiz, A., Ortega, J. ve Martinez-Alvarez, A., 2014, Feature selection by multi-objective optimisation: Application to network anomaly detection by hierarchical self-organising maps, *Knowledge-Based Systems*, 71, 322-338.

Desai, V. S., Crook, J. N. ve Overstreet, G. A., 1996, A comparison of neural networks and linear scoring models in the credit union environment, *European Journal of Operational Research*, 95 (1), 24-37.

Emel, A. B., Oral, M., Reisman, A. ve Yolalan, R., 2003, A credit scoring approach for the commercial banking sector, *Socio-Economic Planning Sciences*, 37, 103-123.

Fogarty, D., 2012, Using Genetic Algoritms for Credit Scoring System Maintenance Functions, *International Journal of Artificial Intelligence & Applications IJAIA*, 3 (6).

Frohlich, H., Chapelle, O. ve Scholkopf, B., 2003, Feature selection for support vector machines by means of genetic algorithms, *15th Ieee International Conference on Tools with Artificial Intelligence, Proceedings*, 142-148.

Garcia, R., Paraiso, E. C. ve Nievola, J. C., 2011, Multiobjective Optimization of Indexes Obtained by Clustering for Feature Selection Methods Evaluation in Genes Expression Microarrays, *Intelligent Data Engineering and Automated Learning - Ideal 2011*, 6936, 353-360.

Ghamisi, P. ve Benediktsson, J. A., 2014, Feature selection of hyperspectral data by considering the integration of Genetic Algorithms and Particle Swarm Optimization, *Image and Signal Processing for Remote Sensing Xx*, 9244.

Guyon, I., Weston, J., Barnhill, S. ve Vapnik, V., 2002, Gene selection for cancer classification using support vector machines, *Machine Learning*, 46 (1-3), 389-422.

Hacıbeyoglu, M., 2012, Bilgi sistemlerinde fark fonksiyonu tabanlı özellik seçme yönteminin geliştirilmesi, *Selçuk Üniversitesi Fen Bilimleri Enstitüsü Doktora Tezi*.

He, J., Zhang, Y. C., Shi, Y. ve Huang, G. Y., 2010, Domain-Driven Classification Based on Multiple Criteria and Multiple Constraint-Level Programming for Intelligent Credit Scoring, *Ieee Transactions on Knowledge and Data Engineering*, 22 (6), 826-838.

Heiat, A., 2012, Comparing performance of data mining models for computer credit scoring, *J. Int. Fin. Econ*, 12 (1), 78-83.

Holland, J. H., 2005, Adaption in natural and artificial systems, *The University of Michigan Press*.

Hu, J., 2009, Personal Credit Rating Using Artificial Intelligence Technology for the National Student Loans, *Iccsse 2009: Proceedings of 2009 4th International Conference on Computer Science & Education*, 103-106.

Huang, B., 2010, Hidden Markov Models - Machine Learning, *Virginia Tech*.

Inman, D., Blind, M., Ribarova, I., Krause, A., Roosenschoon, O., Kassahun, A., Scholten, H., Arampatzis, G., Abrami, G., McIntosh, B. ve Jeffrey, P., 2011, Perceived effectiveness of environmental decision support systems in participatory planning: Evidence from small groups of end-users, *Environmental Modelling & Software*, 26 (3), 302-309.

Jayapriya, J. ve Arock, M., 2015, A Parallel GWO Technique for Aligning Multiple Molecular Sequences, *2015 International Conference on Advances in Computing, Communications and Informatics (Icacci)*, 210-215.

Jones, B., Jenkinson, I., Yang, Z. ve Wang, J., 2010, The use of Bayesian network modelling for maintenance planning in a manufacturing industry, *Reliability Engineering & System Safety*, 95 (3), 267-277.

Kamalloo, E. ve Abadeh, M. S., 2010, An Artificial Immune System for Extracting Fuzzy Rules in Credit Scoring, *2010 Ieee Congress on Evolutionary Computation (Cec)*.

Kamboj, V. K., Bath, S. K. ve Dhillon, J. S., 2016, Solution of non-convex economic load dispatch problem using Grey Wolf Optimizer, *Neural Computing & Applications*, 27 (5), 1301-1316.

Kaplan, O. ve Gozen, G., 2010, "ORACLE DATA MINER" ile mantarların zehirliliği üzerine bir veri madenciliği uygulaması, *İstanbul Teknik Üniversitesi Fen Edebiyat Fakültesi*, Matematik Mühendisliği Programı.

Karakoyun, M. ve Hacıbeyoglu, M., 2014, Biyomedikal Veri Kümeleri İle Makine Öğrenmesi Sınıflandırma Algoritmalarının İstatistiksel Olarak Karşılaştırılması, *DEÜ Mühendislik Fakültesi Mühendislik Bilimleri Dergisi*, 16 (48), 30-41.

Kayacan, E., Ulutas, B. ve Kaynak, O., 2010, Grey system theory-based models in time series prediction, *Expert Systems with Applications*, 37 (2), 1784-1789.

Khemka, A., 2003, A Colloborative Predictive Data Mining Model, *Faculty of University of Missouri Kansas City, Missouri*.

Kisioglu, P. ve Topcu, Y.I., 2011 Applying Bayesian Belief Network Approach to Customer Churn Analysis: A Case Study on the Telecom Industry of Turkey, Expert Systems With Applications, 38(6), 7151-7157.

Kohavi, R. ve John, G. H., 1997, Wrappers for feature subset selection, *Artificial Intelligence*, 97 (1-2), 273-324.

Korayem, L., Khorsid, M. ve Kassem, S. S., 2015, Using Grey Wolf Algorithm to Solve the Capacitated Vehicle Routing Problem, *3rd International Conference on Manufacturing, Optimization, Industrial and Material Engineering (Moime 2015)*, 83.

Kusiak, A., Kern, J. A., Kernstine, K. H. ve Tseng, B. T. L., 2000, Autonomous decision-making: A data mining approach, *Ieee Transactions on Information Technology in Biomedicine*, 4 (4), 274-284.

Lahsasna, A., Ainon, R. N. ve Teh, Y. W., 2008, Intelligent credit scoring model using soft computing approach, *2008 International Conference on Computer and Communication Engineering, Vols 1-3*, 396-402.

Lakka, C., Nikolopoulos, S., Varytimidis, C. ve Kompatsiaris, I., 2011, A Bayesian network modeling approach for cross media analysis, *Signal Processing-Image Communication*, 26 (3), 175-193.

Lal, D. K., Barisal, A. K. ve Tripathy, M., 2016, Grey wolf optimizer algorithm based Fuzzy PID controller for AGC of multi-area power system with TCPS, *2nd International Conference on Intelligent Computing, Communication & Convergence, Iccc 2016*, 92, 99-105.

Leopold, E. ve Kindermann, J., 2002, Text categorization with support vector machines. How to represent texts in input space ?, *Machine Learning*, 46 (1-3), 423-444.

Leung, K., Cheong, F. ve Cheong, C., 2007, Consumer credit scoring using an artificial immune system algorithm, *2007 Ieee Congress on Evolutionary Computation, Vols 1-10, Proceedings*, 3377-3384.

Lin, K. C., Chen, S. Y. ve Hung, J. C., 2015, Feature Selection and Parameter Optimization of Support Vector Machines Based on Modified Artificial Fish Swarm Algorithms, *Mathematical Problems in Engineering*.

Liu, H., Wu, X. ve Zhang, S., 2011, Feature selection using hierarchical feature clustering, *CIKM '11 Proceedings of the 20th ACM international conference on Information and knowledge management*.

Liu, J. X. ve Zhao, Y. H., 2011, The Analysis of Credit Structure Optimization Model Based on Joint Liability, *Proceedings of the 3rd (2011) International Conference on Financial Risk and Corporate Finance Management, Vols 1 and 2*, 592-597.

Liu, Y. J., Cai, Q. R., Luo, Y., Qian, J. ve Ye, F. Y., 2009, Artificial Neural Networks for Corporation Credit Rating Analysis, *2009 International Conference on Networking and Digital Society, Vol 1, Proceedings*, 81-84.

Lloyd, O., 2008, Bayesian belief networks for dementia diagnosis and other applications: a comparison of hand-crafting and construction using a novel data driven technique, *School of Natural Sciences Computing Science and Mathematics*.

Lockamy, A. ve McCormack, K., 2012, Modeling supplier risks using Bayesian networks, *Industrial Management & Data Systems*, 112 (1-2), 313-333.

MacKinnon, M. J. ve Glick, N., 1999, Data mining and knowledge discovery in databases - An overview, *Australian & New Zealand Journal of Statistics*, 41 (3), 255-275.

Manimala, K., Selvi, K. ve Ahila, R., 2011, Hybrid soft computing techniques for feature selection and parameter optimization in power quality data mining, *Applied Soft Computing*, 11 (8), 5485-5497.

Marques, A. I., Garcia, V. ve Sanchez, J. S., 2012, Two-level classifier ensembles for credit risk assessment, *Expert Systems with Applications*, 39 (12), 10916-10922.

Mirjalili, S., Mirjalili, S. M. ve Lewis, A., 2014, Grey Wolf Optimizer, *Advances in Engineering Software*, 69, 46-61.

Mirjalili, S., 2015, How effective is the Grey Wolf optimizer in training multi-layer perceptrons, *Applied Intelligence*, 43 (1), 150-161.

Mitchell, T. M., 1997, Machine Learning *McGraw-Hill Science*, 177-180.

Mittal, N., Singh, U. ve Sohi, B. S., 2016, Modified Grey Wolf Optimizer for Global Engineering Optimization, *Applied Computational Intelligence and Soft Computing*.

Moradi, P. ve Rostami, M., 2015, Integration of graph clustering with ant colony optimization for feature selection, *Knowledge-Based Systems*, 84, 144-161.

Mpofu, T. P. ve Mukosera, M., 2012, Credit Scoring Techniques: A Survey, *International Journal of Science and Research IJSR ISSN*, Online : 2319-7064, 3.358.

Muro, C., Escobedo, R., Spector, L. ve Coppinger, R. P., 2011, Wolf-pack (Canis lupus) hunting strategies emerge from simple rules in computational simulations, *Behavioural Processes*, 88 (3), 192-197.

Murphy, K. P., 2002, Dynamic Bayesian Networks: Representation, Inference and Learning, *Doctor of Philosophy of Computer Science In the GRADUATE DIVISION Of the UNIVERSITY OF CALIFORNIA, BERKELEY*.

Nadkarni, S. ve Shenoy, P. P., 2001, A Bayesian network approach to making inferences in causal maps, *European Journal of Operational Research*, 128 (3), 479-498.

Niu, M. F., Wang, Y. F., Sun, S. L. ve Li, Y. W., 2016, A novel hybrid decomposition-and-ensemble model based on CEEMD and GWO for short-term PM2.5 concentration forecasting, *Atmospheric Environment*, 134, 168-180.

Nosratabadi, H. E., Pourdarab, S. ve Nadali, A., 2011, Credit Risk Assessment of Bank Customers using DEMATEL and Fuzzy Expert System, *Economics and Finance Research*, 4, 255-259.

Oguzlar, A., 2003, Veri Ön İşleme, *Ege Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 21, 73.

Olfati, E., Zarabadipour, H. ve Shoorehdeli, M. A., 2014, Feature Subset Selection and Parameters Optimization for Support Vector Machine in Breast Cancer Diagnosis, *2014 Iranian Conference on Intelligent Systems (Icis)*.

Olson, D. L., Delen, D. ve Meng, Y. Y., 2012, Comparative analysis of data mining methods for bankruptcy prediction, *Decision Support Systems*, 52 (2), 464-473.

Ong, C. S., Huang, J. J. ve Tzeng, G. H., 2005, Building credit scoring models using genetic programming, *Expert Systems with Applications*, 29 (1), 41-47.

Orhan, U. ve Adem, K., 2012, Naive Bayes Yönteminde Olasılık Çarpanlarının Etkileri, *ElektrikElektronik ve Bilgisayar Mühendisliği Sempozyumu*, 723.

Ozdemir, S., 2010, Veri Madenciliği Ders Notları.

Oztemel, E., 2016, Yapay Sinir Ağları Kitabı, *Papatya Bilim Yayınları*.

Pawlak, Z., 2003, A Rough Set View on Bayes' Theorem, *INTERNATIONAL JOURNAL OF INTELLIGENT SYSTEMS*, 18, 487–498.

Pederson, S., 1998, Some statistical aspects of credit scoring, *Dimension Reduction, Computational Complexity and Information*, 30, 264-268.

Perez-Minana, E., Krause, P. J. ve Thornton, J., 2012, Bayesian Networks for the management of greenhouse gas emissions in the British agricultural sector, *Environmental Modelling & Software*, 35, 132-148.

Platt, J. C., Cristianini, N. ve Shawe-Taylor, J., 2000, Large margin DAGs for multiclass classification, *Advances in Neural Information Processing Systems 12*, 12, 547-553.

Sadatrasoul, S., Gholamian, M. ve Shahanaghi, K., 2015, Combination of Feature Selection and Optimized Fuzzy Apriori Rules: The Case of Credit Scoring, *International Arab Journal of Information Technology*, 12 (2), 138-145.

Sahin, S. O., Ulengin, F. ve Ulengin, B., 2004, Using neural networks and cognitive mapping in scenario analysis: The case of Turkey's inflation dynamics, *European Journal of Operational Research*, 158 (1), 124-145.

Savas, S., Topaloglu, N. ve Yılmaz, M., Veri Madenciliği ve Türkiye' deki Uygulama Örnekleri, *İstanbul Ticaret Üniversitesi, Fen Bilimleri Dergisi*, 21, 1-23.

Seker, S. E., 2008, Entropi (Entropy, Dağınım, Dağıntı), *bilgisayarkavramlari.sadievrenseker.com/2008/12/17/entropi-entropy/*.

Seker, S. E., 2013, K Fold Cross Validation (K Katlamalı Çapraz Doğrulama), *http://bilgisayarkavramlari.sadievrenseker.com/2013/03/31/k-fold-cross-validation-k-katlamali-carpraz-dogrulama/*.

Shakarami, M. R. ve Davoudkhani, I. F., 2016, Wide-area power system stabilizer design based on Grey Wolf Optimization algorithm considering the time delay, *Electric Power Systems Research*, 133, 149-159.

Shao, L. S., Zhang, H. Y., Zheng, Y. X. ve Dou, W. H., 2005, Mathematics model and performance evaluation of a scalable TCP congestion control protocol to LNCS/LNAI proceedings, *Grid and Cooperative Computing - Gcc 2005, Proceedings*, 3795, 1054-1065.

Sharma, Y. ve Saikia, L. C., 2015, Automatic generation control of a multi-area ST - Thermal power system using Grey Wolf Optimizer algorithm based classical controllers, *International Journal of Electrical Power & Energy Systems*, 73, 853-862.

Siegmund-Schultze, R., 2004, Mathematicians forced to philosophize: An introduction to Khinchin's paper on von Mises' theory of probability, *Science in Context*, 17 (3), 373-390.

Sonmez, F., 2015, Kredi Skorunun Belirlenmesinde Yapay Sinir Ağları ve Karar Ağaçlarının Kullanımı: Bir Model Önerisi, *http://abmyod.aydin.edu.tr/makaleler/sayi37_38_39_40/kredi-skorunun-belirlenmesinde-yapay-sinir-aglari-ve-karar-agaclarinin-kullanimi-bir-model-onerisi.pdf*.

Sulaiman, M. H., Mustaffa, Z., Mohamed, M. R. ve Abdullah, N. R. H., 2015, An Application of Cuckoo Search Algorithm for Solving Combined Economic and Emission Dispatch Problem, *2015 4th International Conference on Informatics, Electronics & Vision Iciev 15*.

Tsai, C. F. ve Wu, J. W., 2008, Using neural network ensembles for bankruptcy prediction and credit scoring, *Expert Systems with Applications*, 34 (4), 2639-2649.

Turabieh, H., 2016, A Hybrid ANN-GWO Algorithm for Prediction of Heart Disease, *American Journal of Operations Research*, 6, 136-146.

Van Gestel, T., Baesens, B., Suykens, J., Espinoza, M., Baestaens, D. E., Vanthienen, J. ve De Moor, B., 2003, Bankruptcy prediction with Least Squares Support Vector Machine Classifiers, *2003 Ieee International Conference on Computational Intelligence for Financial Engineering, Proceedings*, 1-8.

Van Gool, J., Verbeke, W., Sercu, P. ve Baesens, B., 2012, Credit scoring for microfinance: is it worth it?, *International Journal of Finance & Economics*, 17 (2), 103-123.

Vassilia, P. N. ve Konstantinos, M. G., 2006, Multimodal continuous recognition system for Greek Sign Language using various grammars, *Advances in Artificial Intelligence, Proceedings*, 3955, 584-587.

Waad, B., Ghazi, B. M. ve Mohamed, L., 2013, On the Effect of Search Strategies On Wrapper Feature Selection in Credit Scoring, *2013 International Conference on Control, Decision and Information Technologies (Codit)*, 218-223.

Wang, J. ve Chen, Q. Y., 2007, Next generation of data mining applications., *Interfaces*, 37 (2), 199-201.

Wang, L. ve Li, L. P., 2013, An effective differential harmony search algorithm for the solving non-convex economic load dispatch problems, *International Journal of Electrical Power & Energy Systems*, 44 (1), 832-843.

Wang, Y., Tang, J. ve Cao, W., 2012, Grey Prediction Model-Based Food Security Early Warning Prediction, *Theory and Application*, 2 (1), 13-23.

Warner, H. R., Toronto, A. F., Veasey, L. G. ve Stephenson, R., 1992, A Mathematical Approach to Medical Diagnosis - Application to Congenital Heart-Disease, *M D Computing*, 9 (1), 43-50.

Wedding, D. K., 2005, Discovering knowledge in data, an introduction to data mining., *Information Processing & Management*, 41 (5), 1307-1309.

Winkler, R. L., 2001, Why Bayesian analysis hasn't caught on in healthcare decision making, *International Journal of Technology Assessment in Health Care*, 17 (1), 56-66.

Xue, B., Qin, A. K. ve Zhang, M. J., 2014, An Archive Based Particle Swarm Optimisation for Feature Selection in Classification, *2014 Ieee Congress on Evolutionary Computation (Cec)*, 3119-3126.

Yan, L., Miller, D.J., Mozer, M.C., ve Wolniewicz, R. 2001, Improving prediction of customer behaviour in non-stationary environments. Proceedings of International Joint Conference on Neural Networks, 2258-2263

Yang, Y., Dong, Y., Chen, Y. H. ve Li, C. H., 2014, Intelligent Optimized Combined Model Based on GARCH and SVM for Forecasting Electricity Price of New South Wales, Australia, *Abstract and Applied Analysis*.

Yap, B. W., Ong, S. H. ve Husain, N. H. M., 2011, Using data mining to improve assessment of credit worthiness via credit scoring models, *Expert Systems with Applications*, 38 (10), 13274-13283.

Yaralıoglu, K., 2013, Veri Madenciliği.

Yazici, M., 2011, Combination of Discriminant Analysis and Artificial Neural Network in the Analysis of Credit Card Customers, *European Journal of Finance and Banking Research*, 4.

Yusof, Y. ve Mustaffa, Z., 2015, Time Series Forecasting of Energy Commodity using Grey Wolf Optimizer, *Proceedings of the International MultiConference of Engineers and Computer Scientists 2015*, 1.

EKLER

EK-1 GWO Algoritmasının C# Kodu

```
class GWOFunctions
    {
        #region CustomerList

        private List<CustomerContract> customerList;

        public List<CustomerContract> CustomerList
        {
            get;
            set;
        }

        #endregion CustomerList
        public class TestData
        {
            public String FieldName { get; set; }
            public List<String> FieldList { get; set; }
            public Double TestNote { get; set; }
        }

        private List<TestData> testList;

        public List<TestData> TestList
        {
            get;
            set;
        }
        public List<TestData> TestNode { get; set; }
        public class ID3
        {
            public String NodeName { get; set; }
            protected List<ID3> nNodes;
            public List<ID3> Nodes
            {
                get { return nNodes; }
                set { nNodes = value; }

            }
        }
        public Double[,] testData { get; set; }

        public double[] Create_pos(int dim)
        {
            double[] resultList = new double[dim];

            for (int i = 0; i < dim; i++)
            {
                resultList[i] = 0;
            }

            return resultList;
        }

        public double[,] initialization(int SearchAgents_no, int dim, double ub,
double lb)
        {
```

```
            double[,] Positions = new double[SearchAgents_no, dim];
            Random rnd = new Random();
            if (testData != null && testData.Length > 0)
                for (int i = 0; i < SearchAgents_no; i++)
                {
                    for (int j = 0; j < dim; j++)
                    {
                        Positions[i, j] = testData[i, j] * (ub - lb) + lb;
                    }
                }

            return Positions;

        }
        public Double GetRandomNumber(double minimum, double maximum)
        {
            Random random = new Random();
            return random.NextDouble() * (maximum - minimum) + minimum;
        }
        public double[,] GWO(int SearchAgents_no, int Max_iter, double lb, double
ub, int dim, int funcIndex, Double[,] fTestData)
        {
            // initialize alpha, beta, and delta_pos
            double[] Alpha_pos = Create_pos(dim);
            double Alpha_score = double.PositiveInfinity;
            double[] Beta_pos = Create_pos(dim);
            double Beta_score = double.PositiveInfinity;
            double[] Delta_pos = Create_pos(dim);
            double Delta_score = double.PositiveInfinity;
            Random rnd = new Random();

            // String Sonuc = "";
            double[,] Sonuc = new double[Max_iter, SearchAgents_no];
            testData = fTestData;

            //Initialize the positions of search agents
            double[,] Positions = initialization(SearchAgents_no, dim, ub, lb);
            int l = 0;// Loop counter
            List<double> fitnessList = new List<double>();
            double fitness = 0.0;
            int index = 0;
            while (l < Max_iter)
            {
                for (int i = 0; i < SearchAgents_no; i++)
                {
                    for (int j = 0; j < dim; j++)
                    {
                        if (Positions[i, j] > ub)
                        {
                            Positions[i, j] = ub;
                        }
                        else if (Positions[i, j] < lb)
                        {
                            Positions[i, j] = lb;
                        }
                    }

                    List<double> currentPosition = new List<double>();

                    for (int k = 0; k < dim; k++)
                    {
                        currentPosition.Add(Positions[i, k]);
```

```
                    }
                    fitness = RunBenchmarkFunc(currentPosition, funcIndex);

                    if (fitness < Alpha_score)
                    {
                        Alpha_score = fitness;
                        Alpha_pos = currentPosition.ToArray();
                        //Console.WriteLine(testDataFieldName[i]+" "+"l:  " +
l.ToString() + "\t value: " + Alpha_score.ToString());
                    }
                    else if (fitness > Alpha_score && fitness < Beta_score)
                    {
                        Beta_score = fitness;
                        Beta_pos = currentPosition.ToArray();
                        // Console.WriteLine(testDataFieldName[i] + " " + "l:  "
+ l.ToString() + "\t value: " + Beta_score.ToString());
                    }
                    else if (fitness > Alpha_score && fitness > Beta_score &&
fitness < Delta_score)
                    {
                        Delta_score = fitness;
                        Delta_pos = currentPosition.ToArray();
                        // Console.WriteLine(testDataFieldName[i] + " " + "l:  "
+ l.ToString() + "\t value: " + Delta_score.ToString());
                    }
                    //Sonuc = Sonuc + Delta_score.ToString()+"    ";
                    Sonuc[index, i] = fitness;

                }
                index = index + 1;
                // Sonuc = Sonuc + "\n";
                double a = 2 - l * (2 / Max_iter);

                for (int i = 0; i < SearchAgents_no; i++)
                {
                    for (int j = 0; j < dim; j++)
                    {
                        double r1 = rnd.NextDouble();
                        double r2 = rnd.NextDouble();
                        double A1 = 2 * a * r1 - a; // Equation (3.3)
                        double C1 = 2 * r2; // Equation (3.4)

                        double D_alpha = Math.Abs(C1 * Alpha_pos[j] -
Positions[i, j]); // Equation (3.5)-part 1
                        double X1 = Alpha_pos[j] - A1 * D_alpha; // Equation
(3.6)-part 1

                        r1 = rnd.NextDouble();
                        r2 = rnd.NextDouble();

                        double A2 = 2 * a * r1 - a; // Equation (3.3)
                        double C2 = 2 * r2; // Equation (3.4)

                        double D_beta = Math.Abs(C2 * Beta_pos[j] - Positions[i,
j]); // Equation (3.5)-part 2
                        double X2 = Beta_pos[j] - A2 * D_beta; // Equation (3.6)-
part 2

                        r1 = rnd.NextDouble();
                        r2 = rnd.NextDouble();

                        double A3 = 2 * a * r1 - a; // Equation (3.3)
```

```
                        double C3 = 2 * r2; // Equation (3.4)

                        double D_delta = Math.Abs(C3 * Delta_pos[j] -
Positions[i, j]); // Equation (3.5)-part 3
                        double X3 = Delta_pos[j] - A3 * D_delta; // Equation
(3.5)-part 3

                        Positions[i, j] = (X1 + X2 + X3) / 3;// Equation (3.7)
                    }
                }
                fitnessList.Add(fitness);
                l++;
            }

            for (int i = 0; i < Alpha_pos.Length; i++)
            {
                //   Console.WriteLine(string.Format("{0:E}",
Alpha_pos[i].ToString()) + " ");
            }
            return Sonuc;
        }


        public double RunBenchmarkFunc(List<double> agent, int funcIndex)
        {
            //1     Sphere
            //2     Schwefel
            //3     Rosenbrock
            int n = agent.Count;
            double result = 0;

            switch (funcIndex)
            {
                case 1://1 Sphere

                    for (int i = 0; i < n; i++)
                    {
                        result = result + agent[i] * agent[i];
                    }
                    // each xi is between [-5, 5], you can modify n
                    break;

                case 2://2 Schwefel 2.22

                    for (int i = 0; i < n; i++)
                    {
                        result = result + Math.Abs(agent[i]);
                    }

                    double temp = 1;

                    for (int i = 0; i < n; i++)
                    {
                        temp = temp * Math.Abs(agent[i]);
                    }

                    result = result + temp;
                    break;

                case 3:

                    for (int i = 0; i < n; i++)
```

```
                    {
                        double total = 0;

                        for (int j = 0; j <= i; j++)
                        {
                            total = total + agent[j];
                        }

                        result = result + Math.Pow(total, 2);
                    }

                    break;

                case 4://3 Rosenbrock
                    //  for i=1:(size(x,2)-1)
                    //  val=val+100*(x(i+1)-(x(i))^2)^2+(x(i)-1)^2;

                    for (int i = 0; i < n - 1; i++)
                    {
                        result = result + 100 * Math.Pow((agent[i + 1] -
Math.Pow(agent[i], 2)), 2) + Math.Pow((agent[i] - 1), 2);
                    }
                    //x from[-10 10]
                    break;

            }

            return result;
        }

        public List<double> GetBoundary(int funcIndex)
        {
            double lb = 0, ub = 0;

            switch (funcIndex)
            {
                case 1:
                    lb = -100;
                    ub = 100;
                    break;

                case 2:
                    lb = -10;
                    ub = 10;
                    break;
                case 3:
                    lb = -100;
                    ub = 100;
                    break;
            }

            return new List<double>(2) { ub, lb };

        }

    }
```

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı	**:**	Ali TUNÇ
Uyruğu	**:**	TC
Doğum Yeri ve Tarihi	**:**	Meram / 09.04.1983
Telefon	**:**	05055298223
Faks	**:**	
e-mail	**:**	alitunc42@gmail.com

EĞİTİM

Derece		**Adı, İlçe, İl**	**Bitirme Yılı**
Lise	:	Karatay Lisesi, Karatay, Konya	2000
Üniversite	:	Selçuk Üniversitesi, Selçuklu, Konya	2006
Yüksek Lisans	:	Selçuk Üniversitesi, Selçuklu, Konya	-

İŞ DENEYİMLERİ

Yıl	**Kurum**	**Görevi**
2006-2014	İttifak Holding	Yazılım Geliştirme Mühendisi
2014-2016	Kuveyttürk Katılım Bankası	Kıdemli Yazılım Mühendisi
2016-	Kuveyttürk Katılım Bankası	Takım Lideri

UZMANLIK ALANI

Veri Tabanları Yönetimi, Yazılım Geliştirme, Programlama Dilleri, ERP Sistemleri, Finans Yönetimi, Sınıflandırma, Yapay Zekâ, Makina Öğrenmesi,

YABANCI DİLLER

İngilizce

YAYINLAR

Tunç A, 2014 , Dağıtık Depo Yönetim Sistemi ve Mobil Terminal Erişimi Üzerine Deneyim Paylaşımı, UYMS 2014 Proceedings of the 8th Turkish National Software Engineering Symposium.

Tunç A, Yanar Ö, Ülger I, 2015, AUTOMATED DOCUMENT CATEGORIZATION USING MICROSOFT SQL SERVER FULL TEXT SEARCH AND TF-IDF TERM WEIGHTING, International Conference on Advanced Technology & Sciences, 2th International Conference, ICAT'15 Antalya, Turkey, August 04-07, 2015.

Tunç A., Ülger İ 2016, Veri Madenciliği Uygulamalarında Özellik Seçimi İçin Finansal Değerlere Binning ve Five Number Summary Metotları ile Normalizasyon İşleminin Uygulanması, 18. Akademik Bilişim Konferansı, Adnan Menderes Üniversitesi, Aydın.

Tunç A., Şenal İ., Başçiftçi F. 2016, Normalization with the Five Number Summary on Financial Values By Applying Binning Method and Comparison of Performance, SELÇUK INTERNATIONAL SCIENTIFIC CONFERENCE ON APPLIED SCIENCES – 2016 27-30 September 2016, Antalya / TURKEY.