

TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**DENİZ YOLU ÜZERİNDE GÖÇMEN SAYISI TAHMİNİ VE DEVRİYE BOTU
ROTALAMASI**



DOKTORA TEZİ

Fatma ÇARMAN ÇEVİK

Endüstri Mühendisliği Anabilim Dalı

Tez Danışmanı: Prof. Dr. Tahir HANALİOĞLU

ARALIK 2021

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, alıntı yapılan kaynaklara eksiksiz atıf yapıldığını, referansların tam olarak belirtildiğini ve ayrıca bu tezin TOBB ETÜ Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırlandığını bildiririm.

Fatma Çarman Çevik

ÖZET

Doktora Tezi

DENİZ YOLU ÜZERİNDE GÖÇMEN SAYISI TAHMİNİ VE DEVRİYE BOTU ROTALAMASI

Fatma Çarman Çevik

TOBB Ekonomi ve Teknoloji Üniversitesi
Fen Bilimleri Enstitüsü
Endüstri Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Tahir Hanalioğlu

Tarih: Aralık 2021

Bu çalışmada ilk kez bir milli güvenlik projesi için Türkiye'nin deniz hattındaki göçmen sayısının tahmini ve devriye botu rotalaması problemini ele alıyoruz. Göçmenlerin denizyolu ile taşınması, dünyadaki diğer ülkelerin yanı sıra Türkiye'nin de en büyük endişelerinden biridir. Belgesiz göçmenleri yakalamak için çeşitli araç türlerinin kullanılmasının maliyeti son derece yüksek olabilir. Hem harcama sorununu en aza indirmek hem de doğru bir güvenlik taraması sağlamak için ilgili bölgelerdeki göçmen yoğunlukları da göz önünde bulundurularak optimize edilmelidir. Bu amaca ulaşmak için ilk aşamada deniz yolu güzergahındaki göçmen sayısını tahminini önemli bir problem olarak görüyoruz ve olasılıksal ve olasılıksal olmayan modellerle çalışıyoruz. COVID-19 nedeniyle 2020 ve 2021 yılına ait verilere henüz ulaşamadığımızdan, Türkiye Cumhuriyeti İçişleri Bakanlığı bünyesindeki Sahil Güvenlik Komutanlığı'ndan elde edilen 2016-2019 yılları arası gözlemler kullanılarak 2019 yılı veri setine ilişkin tahminler elde etmek ve fiili gözlemleri karşılaştırmak amacıyla basitten karmaşığa yedi seçkin tahmin yöntemi uyguluyoruz. Ardından, tüm yöntemleri birleştiren Meta Bulanık Fonksiyonlarla Tahmin Kombinasyonu Yaklaşımı öneriyoruz. Nihai değerlendirmelere göre, önerilen yaklaşımın, Türkiye'nin deniz hattında beklenen göçmen sayısı için daha doğru tahmin sonuçları verdiğini

gösteriyoruz ve bu sonuçları devriye planlaması için kullanıyoruz. İkinci aşamada, 2016-2019 yıllarına ait gözlemlerin koordinatlarını kümeleyerek sıcak noktaları belirliyoruz ve bu noktaları ziyaret etmek için karma bir araç rotalama problemini çözüyoruz. Ardından Komutanlığın kaynaklarının sınırlı olduğu durumu ele alıyoruz ve sıcak noktaları kritikliklerine göre önceliklendirmek için bir yaklaşım öneriyoruz. İlk aşamada elde edilen önerilen yöntemin tahmin sonuçlarını sıcak noktaların önem ağırlığı olarak kullanıyoruz. Bu ağırlıkları, tespit gücünü ve aynı anda kat edilen toplam mesafeyi dikkate alan bir yönlendirme formülasyonuna entegre ediyoruz. Biobjektif bir yaklaşımla, problemin karar vericileri tarafından dikkate alınabilecek farklı mesafe ve tespit gücü seviyelerinde alternatif verimli çözümler üretiyoruz. Bu, Türkiye'de maliyetleri düşürecek ve ulusal güvenliği artıracak verimli deniz yolu devriye rotaları oluşturmayı amaçlayan ilk çalışmadır. Bu problem için kümeleme, tahmin ve çok amaçlı optimizasyon tekniklerini entegre eden kapsamlı bir metodoloji sunuyoruz.

Anahtar Kelimeler: Göç, Tahmin, Meta-Bulanık Fonksiyonlar, Devriye Aracı Rotalaması, Çok Kriterli Optimizasyon

ABSTRACT

Doctor of Philosophy

FORECASTING THE NUMBER OF IMMIGRANTS ON THE MARITIME LINE AND PATROL BOAT ROUTING

Fatma arman evik

TOBB University of Economics and Technology
Institute of Natural and Applied Sciences
Industrial Engineering Science Programme

Supervisor: Prof. Dr. Tahir Hanaliođlu

Date: December 2021

This study focuses on the problem of estimating the number of immigrants on Turkey's maritime line and patrol boat routing for the first time for a national security project. Transporting immigrants by sea is one of Turkey's biggest concerns, among other countries in the world. The cost of using various types of vehicles to catch undocumented immigrants can be extremely high. It should be optimised with the concentration of immigrants in the relevant regions in mind, both to minimize the spending problem and to ensure an accurate security screening. In order to achieve this goal, estimating the number of immigrants along the maritime line route in the first stage is seen as a significant problem and possibilistic and non-probabilistic models are studied. Since the data for 2020 and 2021 have not yet been accessed due to COVID-19, seven distinguished forecasting methods are applied from simple to complex to obtain estimates of the 2019 data set and to compare actual observations using dataset between 2016 and 2019 obtained from the Coast Guard Command within the Ministry of Interior of the Republic of Turkey. Then, the Forecast Combination Approach with Meta Fuzzy Functions, which combines all methods, is proposed. According to the final assessments, the proposed approach provides more accurate forecast results for the expected number of migrants on Turkey's maritime line, which

are used for patrol planning. In the second step, we identify the hotspots by clustering the coordinates of the observations from 2016-2019 and solve a mixed vehicle routing problem to visit these hotspots. We then address the situation where the Command's resources are limited and propose an approach to prioritize hotspots based on their criticality. We use the estimation results of the proposed method obtained in the first step as the criticality weight of the hotspots. We integrate these weights into a steering formulation that takes into account the detection power and the total distance traveled simultaneously. With a biobjective approach, we produce alternative efficient solutions at different distance and detection power levels that can be taken into account by the decision makers of the problem. This is the first study in Turkey that aims to create efficient maritime patrol routes that will reduce costs and increase national security. We present a comprehensive methodology for this problem integrating clustering, prediction and multi-objective optimization techniques.

Keywords: Immigration, Forecasting, Meta-Fuzzy Functions, Patrol Vehicle Routing, Multi-Criteria Optimization

TEŐEKKÜR

Çalıőmalarım boyunca deęerli yardım ve katkılarıyla beni yönlendiren hocam Prof. Dr. Tahir HANALIOęLU'na, kıymetli tecrübelerinden faydalandığım TOBBEkonomi ve Teknoloji Üniversitesi Endüstri Mühendislięi Bölümü öğretim üyesi SalihTEKİN'e, Hacettepe Üniversitesi Endüstri Mühendislięi Bölümü öğretim üyesi CerenTUNCER ŐAKAR'a, THK Üniversitesi Endüstri Mühendislięi Bölümü öğretim üyesi Baőak GEVER'e ve destekleriyle her zaman yanımda olan eőim A.Çaękan ÇEVİK ve aileme çok teőekkür ederim.

İÇİNDEKİLER

	<u>Sayfa</u>
ÖZET	iv
ABSTRACT	vi
TEŞEKKÜR	viii
İÇİNDEKİLER	ix
ŞEKİL LİSTESİ	xi
ÇİZELGE LİSTESİ	xii
KISALTMALAR	xiv
SEMBOL LİSTESİ	xv
1. GİRİŞ	2
1.1 Tezin amacı	2
1.2 Literatür araştırması.....	3
2. ZAMAN SERİLERİ ANALİZİ	10
2.1. Zaman serilerinin matematiksel olarak modellenmesi	10
2.2. Zaman serilerinde tahmin	12
3. TAHMİN	15
3.1 Olasılıksal tahmin modelleri	15
3.1.1 Ototegresif Entegreli Hareketli Ortalama	15
3.1.2 Üstel Yumuşatma	16
3.2 Olasılıksal olmayan tahmin modelleri	17
3.2.1 Tekrarlayan Tip-1 Bulanık Fonksiyonlar	17
3.2.1.1 Parçacık Sürüsü Optimizasyonu.....	23
3.2.2 Uyarlanabilir Nöro-Bulanık Çıkarım Sistemleri	25
3.2.3 Yapay Sinir Ağları.....	27
3.2.4 Uzun-Kısa Süreli Bellek.....	32
3.2.5 Meta Bulanık Fonksiyonlar	36
4. KARMA ARAÇ ROTALAMA ALGORİTMALARI	41
4.1. Geleneksel karma araç rotalama modeli.....	41
5. KÜMELEME ALGORİTMALARI	47
5.1 K-Means kümeleme algoritması.....	48
6. GÖÇMEN SAYISI TAHMİNİ VE DEVRİYE BOTU ROTALAMASI	50
6.1 Problemin Tanımlanması	50
6.1.1 Problemin verileri.....	51
6.1.2 Verilerin analizi	55
6.1.3 Problemin varsayımları	61
6.2 Göçmen sayısı tahmini.....	61
6.2.1 Önerilen tahmin yaklaşımı (FCA-MFFs)	62
6.2.1.1 Kıta sahanlığı veri seti üzerinde göçmen sayısı tahmini	65
6.2.1.2 Bölge-1 veri seti üzerinde göçmen sayısı tahmini.....	70
6.2.1.3 Bölge-2 veri seti üzerinde göçmen sayısı tahmini.....	74
6.2.1.4 Bölge-3 veri seti üzerinde göçmen sayısı tahmini.....	78
6.2.1.5 Bölge-4 veri seti üzerinde göçmen sayısı tahmini.....	82
6.2.1.6 Genel değerlendirme ve yorum	86

6.3 Devriye botu rotalaması.....	88
6.3.1 Önerilen devriye botu rotalaması yaklaşımı (MFF-I-MVRP)	89
6.3.1.1 Talep noktalarının belirlenmesi.....	89
6.3.1.2 Senaryo 1 : Geleneksel karma araç rotalama modeli	96
6.3.1.3 Sıcak noktaların önem ağırlıklarının belirlenmesi	98
6.3.1.4 Senaryo 2 : Tespit gücü sınırlarına sahip önem ağırlıklı rotalama modeli	102
6.3.1.5 Genel değerlendirme ve yorum.....	110
7. SONUÇ VE ÖNERİLER	111
KAYNAKLAR	114
EKLER.....	123
ÖZGEÇMİŞ.....	129



ŞEKİL LİSTESİ

Sayfa

Şekil 3.1 : İki girdili 2 kurallı ANFIS	26
Şekil 3.2 : Tipik ileri beslemeli çok katmanlı algılayıcılar	29
Şekil 3.3 : LSTM akış şeması.....	35
Şekil 5.1 : Kümeleme gösterimi	47
Şekil 6.1 : Göçmen yakalanan koordinatların harita üzerinde gösterimi	52
Şekil 6.2 : Bölgelerin gruplandırılması gösterimi	55
Şekil 6.3 : Kıta sahanlığı için haftalık bazda müdahale edilen göçmen analizi	57
Şekil 6.4 : Bölge-1 için haftalık bazda müdahale edilen göçmen analizi.....	59
Şekil 6.5 : Bölge-2 için haftalık bazda müdahale edilen göçmen analizi.....	59
Şekil 6.6 : Bölge-3 için haftalık bazda müdahale edilen göçmen analizi.....	60
Şekil 6.7 : Bölge-4 için haftalık bazda müdahale edilen göçmen analizi.....	60
Şekil 6.8 : Genel karşılaştırma	61
Şekil 6.9 : Kıta sahanlığı verileri.....	68
Şekil 6.10 : Tahmin sonuçlarının karşılaştırılması	68
Şekil 6.11 : Bölge-1 verileri	72
Şekil 6.12 : Tahmin sonuçlarının karşılaştırılması	72
Şekil 6.13 : Bölge-2 verileri	76
Şekil 6.14 : Tahmin sonuçlarının karşılaştırılması.....	76
Şekil 6.15 : Bölge-3 verileri	80
Şekil 6.16 : Tahmin sonuçlarının karşılaştırılması	80
Şekil 6.17 : Bölge-4 verileri	84
Şekil 6.18 : Tahmin sonuçlarının karşılaştırılması	84
Şekil 6.19 : Pilot bölge gösterimi	91
Şekil 6.20 : Küme merkezleri.....	93
Şekil 6.21 : Sıcak noktaların gruplandırılması	94
Şekil 6.22 : Etkin çözümler	106
Şekil 6.23 : Yeni ağırlıklarla elde edilen etkin çözümler	109

ÇİZELGE LİSTESİ

Sayfa

Çizelge 3.1 : İlk parçacık için pozisyonlar	20
Çizelge 6.1 : Enlem-bölge eşleştirmesi	55
Çizelge 6.2 : Devriye botu tipleri	56
Çizelge 6.3 : Kıta sahanlığı $Z_{doğrulama}$ veri seti	69
Çizelge 6.4 : Yöntemlerin fonksiyonlardaki ağırlıkları ve $Z_{doğrulama}$ için RMSE değerleri.....	69
Çizelge 6.5 : Kıta sahanlığı Z_{test} veri seti ve yöntemlerin RMSE değerleri	70
Çizelge 6.6 : Kıta sahanlığı veri seti için mevcut yöntemler ve FCA-MFFs tahmin sonuçları ve RMSE karşılaştırması	70
Çizelge 6.7 : Bölge-1 için $Z_{doğrulama}$ veri seti.....	73
Çizelge 6.8 : Yöntemlerin fonksiyonlardaki ağırlıkları ve $Z_{doğrulama}$ için RMSE değerleri	73
Çizelge 6.9 : Bölge-1 için Z_{test} veri seti ve yöntemlerin RMSE değerleri	74
Çizelge 6.10 : Bölge-1 veri seti için mevcut yöntemler ve FCA-MFFs tahmin sonuçları ve RMSE karşılaştırması	74
Çizelge 6.11 : Bölge-2 için $Z_{doğrulama}$ veri seti.....	77
Çizelge 6.12 : Yöntemlerin fonksiyonlardaki ağırlıkları ve $Z_{doğrulama}$ için RMSE değerleri.....	77
Çizelge 6.13 : Bölge-2 için Z_{test} veri seti ve yöntemlerin RMSE değerleri	78
Çizelge 6.14 : Bölge-2 veri seti için mevcut yöntemler ve FCA-MFFs tahmin sonuçları ve RMSE karşılaştırması	78
Çizelge 6.15 : Bölge-3 için $Z_{doğrulama}$ veri seti.....	81
Çizelge 6.16 : Yöntemlerin fonksiyonlardaki ağırlıkları ve $Z_{doğrulama}$ için RMSE değerleri.....	81
Çizelge 6.17 : Bölge-3 için Z_{test} veri seti ve yöntemlerin RMSE değerleri	82
Çizelge 6.18 : Bölge-3 veri seti için mevcut yöntemler ve FCA-MFFs tahmin sonuçları ve RMSE karşılaştırması	82
Çizelge 6.19 : Bölge-4 için $Z_{doğrulama}$ veri seti.....	85
Çizelge 6.20 : Yöntemlerin fonksiyonlardaki ağırlıkları ve $Z_{doğrulama}$ için RMSE değerleri.....	85
Çizelge 6.21 : Bölge-4 için Z_{test} veri seti ve yöntemlerin RMSE değerleri	86
Çizelge 6.22 : Bölge-4 veri seti için mevcut yöntemler ve FCA-MFFs tahmin sonuçları ve RMSE karşılaştırması.	86
Çizelge 6.23 : Bölgeler ve tahmin yöntemlerinin RMSE değerleri özet tablosu	88
Çizelge 6.24 : FCA-MFFs için hesaplama karmaşıklığı	89
Çizelge 6.25 : Devriye botu verileri	91
Çizelge 6.26 : Kümeleme alternatifleri	92
Çizelge 6.27 : Sıcak noktaların gruplandırılmasına ait bilgiler.....	94
Çizelge 6.28 : Küme merkezleri.....	95

Sayfa

Çizelge 6.29 : Senaryo 1 atamaları.....	97
Çizelge 6.30 : Senaryo 1 çözümleri.....	98
Çizelge 6.31 : Enlem-bölge eşleşmesi.....	100
Çizelge 6.32 : Gruplar ve tahmin değerleri	100
Çizelge 6.33 : Sıcak noktalar ve küme içi geçmiş gözlem verileri	101
Çizelge 6.34 : Sıcak noktalar ve önem ağırlıkları	102
Çizelge 6.35 : Ağırlıklı rotalama çözümleri ve atamaları	107
Çizelge 6.36 : Duyarlılık analizi için yeni ağırlıklar	108
Çizelge 6.37 : Yeni ağırlıklarla elde edilen ağırlıklı rotalama çözümleri	110
ÇizelgeEk.1 : BölgeBazındaHaftalıkYakalananGöçmenVerileri	125

KISALTMALAR

AB:	Avrupa Birliđi
ANFIS:	Uyarlanabilir Nöro-Bulanık Çıkarım Sistemleri
ANN:	Yapay Sinir Ağları
AR:	Otoregresif
ARIMA:	Otoregresif Entegreli Hareketli Ortalama
ARMA:	Otoregresif Hareketli Ortalama
ARP:	Araç Rotalama Problemi
ÇKKV:	Çok kriterli Karar Verme
ES:	Üstel Yumuşatma
FCA-MFFs:	Meta Bulanık Fonksiyonlar Entegreli tahmin Kombinasyonu Yaklaşımı
FCM:	Bulanık C-Ortalama
FIS:	Bulanık Çıkarım Sistemleri
GSP:	Gezgin Satıcı Problemi
HES:	Holt-Winters Üstel Yumuşatma
KARP:	Karma Araç Rotalama Problemi
KDARP:	Karma Devriye Aracı Rotalama Problemi
KV:	Karar Verici
LSTM:	Uzun Kısa Süreli Bellek
MA:	Hareketli Ortalama
MAPE:	Ortalama Mutlak Yüzde Hatası
MFF-I-MVRP:	Meta Bulanık Fonksiyonlar Entegreli Karma Araç Rotalama Yaklaşımı
NNs:	Sinir Ağları
NP:	Non Polynomial
PSO:	Parçacık Sürüsü Optimizasyonu
RMSE:	Kök Ortalama Kare Hatası
RNN:	Yinelemeli Sinir Ağları
R-T1FFs:	Tekrarlayan Tip-1 Bulanık Fonksiyonlar
SARIMA:	Mevsimsel Otoregresif Entegreli Hareketli Ortalama
SES:	Simple Üstel Yumuşatma
T1FFs:	Tip-1 Bulanık Fonksiyonlar
TW-MFF-I-MVR:	Zaman Pencere ve Meta Bulanık Fonksiyonlar Entegreli Karma Araç Rotalama Yaklaşımı
ZP-ARP:	Zaman Pencere Araç Rotalama Problemi
ZP-KARP:	Zaman Pencere Karma Araç Rotalama Problemi

SEMBOL LİSTESİ

Bu çalışmada kullanılmış olan simgeler açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	Açıklama
t	Zaman
p	Otoregresif parametresi
d	Hareketli ortalama parametresi
q	Otoregresif entegreli hareketli ortalama parametresi
φ, L, θ	ARIMA katsayıları
ε_t	Bir zaman serisinin bozulma terim değerleri
α, β, γ	Düzeltilme sabiti
μ	Üyelik derecesi
σ	Standart sapma
\otimes	Vektörel çarpım
ϕ	LSTM katsayısı
\in	Elemanıdır
\forall	Her
c	Küme sayısı
p_{best}	Bireysel en iyi
g_{best}	Global en iyi
v	Küme merkezleri
σ^2	Varyans
C	Sabit bir sayı
r_1, r_2	Rassal rakamlar
p, d, q	ARIMA parametreleri
L_t	Basit üstel düzeltme parametresi
S_t	Mevsimsellik
T_t	Trend
F_{t+p}	Tahmin
\hat{Y}	Tahminler
f_i	ANN çarpanı
w_i	Ateşleme gücü
\bar{w}_i	Normalize ateşleme gücü
c_1, c_2	İvme katsayıları

1. GİRİŞ

Türkiye, jeopolitik konumu itibariyle doğu ve batı ülkeleri arasında göçmenlerin kullandığı kritik rota üzerinde yer almaktadır. Göç kavramı sebep olduğu etkilerine bağlı olarak hem iç huzur hem de savunma sanayi kaynaklarının verimli kullanımı için tehdit oluşturabilmektedir. Örneğin kaçakçılık amacı ile yapılan kısa süreli ve geri dönüşlü göçler iç güvenliğe zarar veren niteliktedir. Diğer tip göçler ise mevcut sosyolojik zararlarına ilaveten tespitleri için yapılan devriyelere bağlı olarak devlet kaynaklarının harcanmasına ve ek maliyetlere neden olmaktadır. Bu yönüyle göçler yalnızca demografik rolleri ile sınırlı olmayıp iç güvenlik, ekonomi, sağlık, kültür ve politika üzerine etkisi sebebiyle de incelenmesi gereken bir problemdir.

Göçlerin gerçekleştiği bölgeye göre de analizi karmaşıklaşabilmektedir. Deniz yolu kullanılarak gerçekleşen göçlerde Türkiye'nin yanı sıra komşu ülkeleri de ilgilendiren siyasi süreçler söz konusudur. Türkiye'nin kendi kıta sahanlığı içinde veya uluslararası sular içinde kalan göçlere müdahale hakkı olmakla birlikte komşu ülkelerin kıta sahanlığına ilerlemiş bir göçmene müdahale hakkı bulunmamaktadır. Ek olarak deniz yolu ulaşımının doğası gereği kullanılabilir rota alternatifi sınırsızdır. Bu da deniz yolu üzerinde sistematik bir devriye planlanmasını ve göçmen tespitini zorlaştırmakta ve düzensiz devriyelere bağlı olarak katlanılan maliyetleri artırmaktadır.

Deniz devriye planlaması, deniz yoluyla yasadışı göçle karşı karşıya kalan ülkeler için önemli bir sorundur. Deniz hattında dikkatli seyir ve kritik koordinatların kontrolü, yasadışı göçle mücadelenin önemli bir bileşenidir. Devriye botları bu amaç için yaygın olarak kullanılmaktadır ve verimli kullanımları optimal rota planlarını gerektirmektedir.

1.1 Tezin amacı

Bu çalışmanın amacı, Türkiye Cumhuriyeti İçişleri Bakanlığı bünyesindeki Sahil Güvenlik Komutanlığı'nın ihtiyati tedbirler alarak daha fazla göçmen yakalayabilmesi için 2019 yılının ilk 26 haftasında beklenen göçmen sayısını tahmin etmek ve buna bağlı deniz hattı devriye planlaması yapmaktır.

1.2 Literatür araştırması

Literatür araştırması göçmen sayısı tahmini ve karma devriye aracı rotalaması olmak üzere iki ana başlık altında sunulmuştur.

Göçmen sayısını tahmin etmek, zaman serisi tahminlerinin bir dalıdır. Nüfus artışı veya azalışını ve demografik etkileri tahmin etmek yaygın bir araştırma konusudur. Literatürde beklenen göçmen sayısını, göçe bağlı olarak gelecekteki nüfusu ve göçü etkileyen faktörleri tahmin etmeye yönelik çeşitli çalışmalar bulunmaktadır.

Willekens (1994) ve Kupiszewski (2002)'ye göre, uluslararası göçü tahmin etmedeki ana zorluklar şunlardır: (a) süreçlerin doğasında rastlantısallık ve zorlukla öngörülebilecek faktörlere duyarlılıkları, (b) ülkeler ve zamanlar arasında tutarlı göç tanımları eksikliği, (c) kapsamlı göç teorilerinin eksikliği ve (d) kısa zaman serileri de dahil olmak üzere veri eksikliği veya eksik veri. Bununla birlikte, belirsizlik altındaki göç tahminleri, özellikle gelişmiş ülkelerde güvenilir nüfus tahminleri elde etmek için çok önemlidir. Bu sorunlara kısmi bir çözüm olarak Willekens (1994), uzmanlık bilgisinin göç tahminlerine resmi olarak dahil edilmesini önerir. Bu nedenle çalışmada uzmanlık bilgisi ve sayısal verilerin birleştirilmesiyle ilgili doğal bir metodoloji olan Bayesian tahmin yaklaşımı kullanılır. Uzman kararı, önceki olasılık dağılımları ile temsil edilir ve olasılık fonksiyonuna yansıyan verilerle Bayes teoremi aracılığıyla birleştirilir. Bijak (2006), uluslararası göçle ilgili tahminlerin seçilmiş teorik temellerini özetlemeye çalışır. Alho vd. (2006), 18 Avrupa ülkesinde 2050 yılına kadar nüfus için olası tahminlerin sonuçlarını sunar. Bijak ve Wisniowski (2010), seçilen göç ülkelerindeki göçmen sayısının 2025 yılında Bayesian Tahmin Metodu yardımıyla tahmin edilmesini ele alır. Martineau (2010)'a göre göç krizini durdurmanın en iyi yolu göç başlamadan müdahale etmektir. 170 ülke hakkında bilgi içeren bir veri seti kullanılarak basit bir ikili lojistik modeli oluşturulur. Abel vd. (2013) İngiltere'ye göç ve 2060'a kadar göç konusundaki belirsizliğin tahminini araştırır. Cappelen vd. (2015) tarafından, Norveç'te gelir değişkenleri için üç farklı alternatif belirlenir ve 2100 yılına kadar brüt göç için üç farklı tahmin sonucu elde edilir. Raymer ve Wisniowski (2018), İngiltere'deki göçmen sayısını yaş ve cinsiyete dayalı olarak tahmin etmek için Bayes teoremi çerçevesinde hiyerarşik bir modeli test ederek uluslararası göçün daha iyi tahmin modellerine olan ihtiyacını tartışır. Wicke vd. (2019), Suriye iç savaşı ve göçü temel alarak bu tür sosyo-politik olayların nasıl meydana gelebileceğini ele alan

senaryolar üzerinde çalışır. Suriani vd. (2019), ele aldığı model ile göçmen sayısının önceden tahmin edilebilmesinin ev sahibi ülkenin göçmenlere hazırlıklı olmasına olanak sağladığını belirtir. Model ayrıca göçmen nüfus artışını etkileyen çeşitli girdi faktörlerini de dikkate almakta ve 2017-2022 yılları arasındaki göçmen sayısını tahmin etmektedir.

Göç tahmininde belirsizlik olduğu kadar, belirsizliğin tahminlere nasıl dahil edileceği de önemli bir konudur. Belirsizlik kavramı, incelenen sürecin belirsizliğini veya rastlantısallığını ifade eder. Bu belirsizlikler üç ana başlıkta özetlenebilir (Bijak 2010). Birincisi, gelecekteki olaylarla ilgili doğal belirsizliktir. Göç tahmininde bir miktar hata her zaman kaçınılmazdır, çünkü geleceğe ilişkin çıkarımlar belirsizlik altında yapılır (Bijak 2010). İkinci belirsizlik kaynağı, mevcut göç verilerinin genellikle yanlış, tutarsız ve eksik olmasıdır. Farklı ülkelerden göç verilerinin kaynakları genellikle farklı tanımlara dayanmaktadır (Raymer vd., 2013). Uluslararası göç akışlarının kesin boyutunu ölçmek zordur; göçmenleri kaydetmek için kullanılan veri toplama sistemleri genellikle düzeltilmesi gereken önyargılı ve yanlış tahminler üretir (Disney, 2014 ve Wisniowski, 2013). Üçüncü belirsizlik kaynağı tahmin modellerinin kendisinden gelir. Farklı modellerin aynı verilere uygulanması, tahminlerin belirsizliğinin farklı değerlendirmeleri de dahil olmak üzere farklı tahminler üretebilir. Mükemmel bir model yoktur ve hangi modelin uygulanacağını seçilmesi bir yargı meselesidir. Çeşitli rakip modellerden tahminler resmi ölçütler kullanılarak birleştirilirse, ek model hakkındaki belirsizlik ortaya konmuştur (Bijak ve Wisniowski 2010). Sonuç olarak, herhangi bir göç tahmininde öncelikle tam olarak anlaşılması ve daha sonra ampirik bir analizde dikkate alınması gereken çok sayıda belirsizlik olduğu açıktır. Uzmanlar tahminlerin geliştirilmesinde kilit bir rol oynamaktadır, ancak tahminin başarısı seçilen yaklaşıma bağlıdır (Lutz ve Goldstein 2004). Örneğin, uzmanın rolü, tahmin modelini seçmek ve temeldeki veri kaynaklarını seçmek veya modelde açıkça bir parametre olarak yer alan uzman görüşü sağlamakla sınırlı olabilir.

Literatürdeki bu çalışmaların geleneksel yöntemlerle göçü etkileyen faktörleri araştırmak ve gelecekteki nüfus yoğunluğunu belirlemek amacıyla yapıldığı görülmektedir. Bununla birlikte, son zamanlarda zaman serisi veri setlerini tahmin etmek için çeşitli yöntemler geliştirilmiştir. Bunlara zaman serisi tahmin yöntemleri denir ve iki kategoriye ayrılabilir: olasılıklı ve olasılıklı olmayan yöntemler. İki temel

grup altında toplanan her yöntemin uygulandıkları veri kümesine göre birbirine göre güçlü ve zayıf yönleri bulunmaktadır.

Geleneksel yöntemler olarak da adlandırılan olasılıksal veya stokastik yöntemler, zaman serileri hakkında bazı varsayımlarda bulunur. Durağanlık, olasılıklı zaman serisi tahmin yöntemlerinde önemli bir varsayımdır. Bu varsayım, zaman serilerinin sabit bir ortalama, varyans ve kovaryans fonksiyonuna sahip olduğunu belirtir (Tak vd., 2018).

ARIMA, literatürde özellikle sosyal bilimlerde (Ma 2020) yaygın olarak kullanılan ve Box ve Jenkins (1970) tarafından geliştirilen olasılıksal bir zaman serisi modelleme yöntemidir. ARIMA, incelenen zaman serilerinin doğrusal süreçlerden üretildiğini varsayar. ARIMA'nın tahmin sonuçları girdi değişkenlerinin değerlerinden ve hata terimlerinden türetildiğinden, tahminin herhangi bir temel model veya ilgili denklem varsayması gerekmez (Ma, 2020). ARIMA, kısa vadeli tahmin yeteneğinde genellikle daha karmaşık yapısal modellerden daha iyi performans gösterir. Gelişmiş tahmin modelleri ARIMA'yı kullanır çünkü ARIMA modelleri tahmin performansı açısından sonuçlar verir (Matamoros vd., 2020). Bununla birlikte, ARIMA “geriye döndüktür” ve model tanımlama tekniği öznel, bu da seçilen modelin güvenilirliğinin tahmin edicinin deneyimine ve seçimine bağlı olmasına neden olabilir. Ayrıca bir zaman serisi lineer olmayan bir yapıya sahipse ARIMA modelleri bununla başa çıkamaz (Tak vd., 2018).

Bir başka olasılıksal tahmin modeli, 1950'lerin sonlarında önerilen Üstel Yumuşatma (ES) (Holt, 1957; Brown, 1959; Winters, 1960) yöntemidir. ES yönteminin mekanizması temel olarak zaman serilerini yumuşatmaktır. Bir zaman serisinin stokastik bir eğilimi varsa, bu zaman serisinin problemlerini tahmin etmek için ES yöntemini uygulamak pratiktir. ES yöntemlerinin en büyük avantajı, uygulanmasının basit, sezgisel ve anlaşılmasının kolay olmasıdır. Ek olarak, veri gereksinimleri minimumdur, bu da ES'yi gerçek zamanlı uygulamalar için uygun hale getirir. Ancak yöntemin en büyük dezavantajı modelle ilgili temel önermesinden kaynaklanmaktadır. Uygulanabilmesi için, zaman serisinin düzeyinin sabit bir düzeyde dalgalanması veya zaman içinde kademeli olarak değişmesi gerekir. Zaman serileri belirgin bir trende sahip olduğunda, tahmin için iyi performans göstermezler (Peng vd. 2008).

Uygulamalarda karşılaşılan veri kümeleri genellikle durağanlık veya doğrusallık varsayımını karşılamamaktadır. Bu nedenle, literatür üç ana alternatife dayanır. Birincisi, gelecekteki verileri simüle etmek için geçmiş verilere bir model uydurmaktır. İkincisi, en uygun tahmin modelini seçmektir. Son olarak üçüncüsü, geleneksel zaman serisi tahmininin aksine, zaman serisinin özelliklerine dayanan bir dizi aday yöntemin ortalamasının alınmasıdır (Li vd. 2020).

Son yıllarda birçok araştırmacı, doğrusal olmayan zaman serisi verileri veya durağan olmama durumuyla başa çıkmak için alternatif tahmin yöntemleri olarak olasılıksal olmayan tahmin yöntemleri üzerinde çalışmaktadır. ANFIS, R-T1FFs, ANN ve LSTM yaklaşımları olasılıksal olmayan tahmin yöntemlerine örnek olarak sayılabilir.

Jang (1993) tarafından önerilen ANFIS, Sinir Ağları (NN'ler) ve Bulanık Çıkarım Sistemlerinin (FIS) öğrenme yeteneklerini birleştiren bir nöro-bulanık tekniktir (Çelikyılmaz ve Türken 2009). NN'leri FIS ile birleştirmenin iki büyük faydası vardır. İlk olarak, yöntem veriye dayalıdır, bu nedenle temel modelin formu üzerinde sınırlayıcı varsayımlar yapmaz. İkinci olarak, bireysel kullanım durumlarının bazı dezavantajlarının üstesinden gelir (Atmaca vd. 2001). ANFIS, doğası gereği çeşitli alanlarda birçok sistemin tek tek veya diğer yöntemlerle hibridize edilerek modellenmesi ve tanımlanması için kullanılmaktadır. Örneğin, Barak ve Sadegh (2016), çalışmasında ARIMA ve ANFIS hibrit modelleri uygulanır. Karaboga ve Kaya (2020), İran'dan ve Rusya'dan Türkiye'ye gelen yabancı ziyaretçi sayısını tahmin etmek için ANFIS'i kullanır ve çalışmada tartışılan diğer yöntemlere göre daha iyi sonuçlar verdiğini gözlemler.

Olasılıksal olmayan bir diğer yöntem ise Türkşen (2008) tarafından tanıtılan Tip-1 Bulanık Fonksiyonlar (T1FFs) yöntemidir. Ardından Tak (2018), T1FFs ile Otoregresif ve Hareketli Ortalama (ARMA) modelini aynı algorithmada birleştiren gelişmiş bir model önermektedir. R-T1FFs olarak adlandırılan bu modern yöntem, varsayımdan bağımsız bir yaklaşımdır. Ek olarak, Hareketli Ortalama (MA) modelinin katkısı nedeniyle R-T1FF'nin tahmin yeteneği T1FFs'den daha iyidir. Ayrıca, T1FFs kural tabanlı bir sistem değildir. Kuralların tespiti önemli bir problem olduğundan, R-T1FFs ve ANFIS gibi FIS tabanlı yöntemlere göre avantajlı olabilir.

Diğer olasılıksal olmayan tahmin yöntemleri ise yapay zeka ve derin öğrenme algoritmalarıdır. En iyi bilinen yapay zeka algoritmalarından biri ANN'dir (McCulloch

ve Pitts 1943). ANN genellikle NN'ler olarak adlandırılır ve temelde hayvan beyinlerini oluşturan biyolojik sinir ağlarından belli belirsiz ilham alan bilgi işlem sistemleridir (Chen vd. 2019). ANN'nın tahmini, geniş gözlemler yapmaya ve daha sonra bütünü potansiyel kısmını çıkarmak için orijinal verilerden elde edilen sonuçlara dayanmaktadır. ARIMA modelinin aksine doğrusal olmayan problemlerin çözümünde oldukça etkilidir (Ma 2020). ANN, bireysel veya diğer yöntemlerle melezleştirilmiş olarak, genellikle sosyal, ekonomik, mühendislik, döviz, hisse senedi sorunları vb. ile ilgili tahminlerde kullanılır (Tseng vd. 2002; Zhang 2003; BuHamra vd..2003; Jain ve Kumar 2007; Eğrioglu vd. 2009; Uslu vd. 2010). ANN'nın karmaşık veriler üzerinde doğrusal olmayan analiz yapma yeteneği yüksektir (Ahmad vd. 2014). Ayrıca, iyi bir öğrenme için yeterli olmayan kısa bir veri seti söz konusu olduğunda ANN, ANFIS'e göre avantajlı olabilir (Atmaca vd. 2001).

Diğer bir tahmin yöntemi ise derin öğrenme tabanlı LSTM'dir. LSTM (Hochreiter ve Schmidhuber 1997) uzun süreli bağımlılıkları öğrenebilir. Bu açıdan NN'lerin dezavantajını ortadan kaldırır. Diğer yöntemlerden farklı olarak, geri bildirim bağlantısı, geçmiş verilerin ve güncel verilerin geriye yayılması yoluyla gelişme eğilimlerinin bulunmasını kolaylaştırır (Ma 2020). LSTM, Çeşitli dizi tahmini ve dizi etiketleme görevlerine başarıyla uygulanır (Sak vd. 2014). LSTM algoritması yakın zamanda, Namini vd. (2018), Wang vd. (2020); Chimmula ve Zhang (2020); Shahid vd. (2020) ve Elsheikh (2021) tarafından tahmin problemlerinde başarıyla uygulanır.

Olasılıksal ve olasılıksal olmayan tahmin yöntemleri, daha iyi performans sonuçları vermek için birbirlerinin dezavantajlarını ortadan kaldırmak üzere yıllar içinde geliştirilir. Ancak, son zamanlarda en iyisi olarak kabul edilen tek bir yöntem kullanmak yerine yöntemleri hibritleştirmek daha etkili hale gelir. Bu hibritleme ile her yöntemin avantajlarından yararlanarak tutarlı tahminler elde edilmesini sağlanabilir. Örneğin, Tak (2018) tarafından Meta Bulanık Fonksiyonlar (MFFs), Tak (2020) tarafından Meta Bulanık İndeks Fonksiyonları, Tak ve Gok (2020) tarafından Meta-Olasılıklı İndeks Fonksiyonları ve Tak (2021) tarafından Meta-Olasılıklı Bulanık Fonksiyonlar, farklı yöntemleri bir araya getirerek tahmin doğruluğunu artırmanın mümkün olduğunu göstermektedir. Böylece tahmin sonuçlarının performansının yanı sıra güvenilir ve sağlam oldukları gözlemlenir. Güçlü ve zayıf yönleri yukarıda detaylı olarak açıklanan tahmin yöntemlerinin birleştirilmesiyle, bu

mevcut yöntemlerin bireysel kullanımına göre daha doğru ve daha iyi performans sonuçları elde edildiğine dikkat çekilmektedir.

Karma Devriye Aracı Rotalama Problemi (KDARP), Araç Rotalama Probleminin (ARP) özel bir dalıdır. Dantzig ve Ramser (1959) tarafından Gezgin Satıcı Probleminin (GSP) bir genellemesi olarak tanıtılır ve klasik ARP'nin özel bir durumu olduğu için NP-Zor kombinatoriyal optimizasyon problemi olarak ele alınır. Yöneylem araştırması, lojistik ve taşımacılık alanındaki çalışmalarda sıklıkla karşılaşılmaktadır (Toth ve Vigo 2014, Braekers vd. 2016). Bu, problemin boyutu arttıkça, hesaplama karmaşıklığının katlanarak arttığı anlamına gelir. Orta büyüklükteki ARP problem verilerinde bile çok sayıda değişken kullanılabilir. ARP'nin NP-Zor olması nedeniyle, düğüm sayısının fazla olduğu durumlarda makul hesaplama süresinde optimum turları verecek kesin bir algoritma garanti edilemez. (Codeau vd. 2006). Bu nedenle literatürde bu tür problemlerin çözümü için kesin yöntemler ve daha kısa çözüm süresine sahip sezgisel yöntemler önerilmektedir.

Literatürde devriye aracı rotalama konusunda çeşitli çalışmalar bulunmaktadır. Chawathe (2007), minimum devriye rotası uzunluğu ile sıcak noktaların kapsamını en üst düzeye çıkarmak için devriye rotalarını planlama sorununu ele alır. Quttineh vd. (2013) bir askeri uçak filosu için görev planlama problemini ele alır. Murray ve Park (2013) giderek daha değerli bir askeri kaynak haline gelen İnsansız Hava Araçları (İHA) için görev planlama problemini bir araç rotalama problemi olarak ele alır. Kuo vd. (2013), verileri coğrafi kodlama, sıcak noktaları belirleme, en iyi devriye rotalarını düzenleme ve Kolej Karakolu Polis Departmanından alınan verileri kullanarak etkinliği tahmin etme üzerinde çalışır. Chen vd. (2015), Bayes stratejisine ve Karınca Kolonisi algoritmasına dayalı olarak yol ağı için yürüyen bir polis devriyesi yönlendirme stratejisi sunar. Çalışmada Bayes Karınca Devriye Stratejisi, geçmiş suç haritası ve suç tahmin verileri üzerinden belirli yol bölümlerinin sıcak nokta olarak tanımlandığı yol ağı dikkate alınarak geliştirilir. Chen vd. (2017), polis devriyesinin zorluklarının üstesinden gelmek için devriye yönlendirme stratejileri için bir dizi kılavuz önermektedir. Schermer vd. (2019), Drone'larla Araç Yönlendirme Problemini inceler. Luo vd. (2019), bir kentsel yol sisteminde Drone'larla Trafik Devriyesi Yönlendirme Problemini ele alır. Dewinter vd. (2020), kıt polis kaynaklarını yönetmek ve acil müdahale sürelerini en aza indirmek için verimli bir Dinamik Devriye Yönlendirme Problemi üzerinde çalışır. Saint-Guillain vd. (2020), Brüksel Polis

Departmanının Polis Devriyesi Yönlendirme Problemini çözmek için uygulamalı bir yaklaşım sunar ve problemi Rastgele İsteklerle Zamana Bağlı Statik ve Stokastik Araç Yönlendirme Problemi olarak ele alır. Cardoso vd. (2021), askeri devriyelerin yönlendirilmesi ile ilgilenir. Çalışmanın amacı, sınır gözetimini en etkin şekilde desteklemek için askeri araç devriyelerine yönlendirme sağlamaktır. Timothy ve Chircop (2021), Tam Kapsamlı Karakol Botu Çizelgeleme Probleminin hesaplama karmaşıklığını analiz eder. Bu kombinatorial optimizasyon probleminin deniz sınırlarının korunması ve gözetim operasyonları için önemli etkileri vardır.



2. ZAMAN SERİLERİ ANALİZİ

Zaman serileri, zaman sırasına göre dizilen bir veri noktaları dizisidir. En sık olarak, bir zaman serisi, zaman içinde art arda eşit aralıklarla yerleştirilmiş noktalarda alınan bir sekanstır. Dolayısıyla, ayrık zamanlı bir veri dizisidir.

Zaman serisi modellemesi, serinin doğal yapısını tanımlayan uygun bir model geliştirmek için bir zaman serisinin geçmiş gözlemlerini titizlikle toplamak ve incelemektir. Bu model daha sonra seriler için gelecekteki değerler üretmek, yani tahminler yapmak için kullanılır. Böylece zaman serisi tahmini, geçmişini anlayarak geleceği tahmin etme eylemi olarak adlandırılabilir (Raicharoen vd., 2003). İşletme, ekonomi, finans ve mühendislik vb. gibi birçok pratik alanda zaman serileri tahmininin vazgeçilmez önemi nedeniyle temel zaman serisine uygun bir modele uymaya özen gösterilmelidir (Tong 1983, Zhang 2003). Başarılı bir zaman serisi öngörüsünün uygun bir modellemeye bağlı olduğu açıktır. Tahminlerin performansını iyileştirmek adına verimli modellerin geliştirilmesi için araştırmacılar tarafından uzun yıllar boyunca çaba gösterilir. Sonuç olarak, çeşitli önemli zaman serisi tahmin modelleri literatürde geliştirilir.

Zaman serisi analizi ise, Hipel ve Mcleod (1994) tarafından bir zaman serisini uygun bir modele yerleştirme prosedürü olarak adlandırılır. Bu analiz, serinin yapısını anlamaya çalışan ve gelecek tahmini için yararlı olan yöntemleri içerir. Zaman serisi analizi, zaman içinde alınan veri noktalarının hesaba katılması gereken bir iç yapıya (otokorelasyon, trend veya mevsimsel değişiklik gibi) sahip olabileceği gerçeğini görmemizi sağlar. Uygulamada belirli bir zaman serisine uygun bir modelleme yapılır ve karşılık gelen parametreler bilinen veri değerleri kullanılarak tahmin edilir.

Bir zaman serisini analiz etmenin birkaç olası amacı vardır. Bu hedefler; zaman serisini faktörlerine ayırmak, zaman serileri arasındaki ilişkiyi belirlemek, kontrol amacı ve en yaygın amaç olan geleceğe yönelik tahmin yapmaktır.

2.1. Zaman serilerinin matematiksel olarak modellenmesi

Bir zaman serisi, tipik olarak art arda zamanlarında ölçülen sıralı bir veri noktaları kümesidir. Matematiksel olarak, $x(t)$, $t = 0, 1, 2, \dots$ vektörleri kümesi olarak tanımlanır, burada, t geçen süreyi temsil ederken, $x(t)$ değişkeni rastgele bir değişken olarak kabul

edilir. Böylece zaman serisindeki veriler uygun bir kronolojik sıraya göre düzenlenmiş olur (Adhikari ve Agrawal 2013).

Tek bir değişkene ait kayıtları içeren bir zaman serisi tek değişkenli olarak adlandırılır. Birden fazla değişken dikkate alındığında, çok değişkenli olarak adlandırılır. Bir zaman serisi sürekli veya kesikli olabilir, sürekli bir zaman serisi gözlemleri her zaman ölçülür. Örneğin sıcaklık okumaları, nehrin akışı, kimyasal işlem konsantrasyonları vb. sürekli bir zaman dizisi olarak kaydedilebilir. Öte yandan belirli bir popülasyonda şehir, bir şirketin üretimi, iki farklı para birimi arasındaki döviz kurları kesikli zaman serilerini temsil edebilir. Genellikle kesikli zaman serilerinde ardışık gözlemler, saatlik, günlük, haftalık, aylık veya yıllık zaman ayrımları gibi eşit aralıklı zaman aralıklarında kaydedilir. Hipel ve McLeod (1994) çalışmasında belirtildiği gibi, kesikli bir zaman serisinde gözlemlenen değişkenin, gerçek sayı ölçeği kullanılarak sürekli bir değişken olarak ölçüldüğü varsayılmaktadır. Ayrıca, sürekli bir zaman serisi, verileri belirli bir zaman aralığında bir araya getirerek kolayca kesikli bir birine dönüştürülebilir.

Zaman serisi verileri çeşitli faktörlerin etkisi altındadır ve bu faktörlerin farklı şiddet ve yöndeki hareketine bağlı olarak dalgalanırlar. Zaman serilerindeki dalgalanmaların dört ayrı tür hareketin aynı anda ve birlikte gösterdikleri etkiden meydana geldiği kabul edilir. Zaman serisinin bileşenleri olarak adlandırılan bu hareketler aşağıdaki şekilde tanımlanır:

- a) *Trend*: Zaman serisinde yer alan değerlerin uzun bir dönem içinde gösterdikleri artış veya azalışı ifade eder. Zamana göre ortalamalardaki değişim, verilen serideki bir eğilimin kanıtıdır, ancak zaman serisindeki trendi tespit etmek için daha resmi testler vardır.
- b) *Döngüsel (Konjonktür) dalgalanmalar*: Zaman serileri, sıcaklıktaki günlük değişimler gibi diğer bazı fiziksel nedenlerden dolayı sabit bir periyotta döngüsel dalgalanmalar sergiler. Döngüsel dalgalanmalar, ölçülebilir bir devirde değişen ve mevsimsel olmayan bir bileşendir. Bazen seri sabit bir periyodu olmayan ancak bir dereceye kadar tahmin edilebilir olan salınım sergiler. Örneğin, iş dünyasından etkilenen ekonomik veriler yaklaşık 5 ila 7 yıl arasında değişen bir süreye sahiptir. Konjonktür dalgalanmalar doğada periyodiktir ve kendilerini dört aşamalı (i) Zirve, (ii) Durgunluk,

(iii) Düşüş ve (iv) İyileşme olan iş döngüsü gibi tekrarlar. Döngüsel dalgalanmalar seriden trendin, mevsimsel dalgalanmaların ve tesadüfi hareketlerin çıkarılmasından sonra geriye kalan kısım olarak tanımlanabilirler.

- c) *Mevsimlik dalgalanmalar*: Zaman serilerinde kısa dönem içerisinde görülen dalgalanmalara mevsim dalgalanmaları denir. Zaman serisi verilerinin çoğu, satış ve sıcaklık değerleri gibi mevsimsel bir değişiklik göstermektedir. Bu tür bir varyasyonun anlaşılması kolaydır ve mevsimsellikten arındırılmış veriler elde etmek için mevsimsellik kolayca ölçülebilir ve verilerin çıkarılabilir. Örneğin, işsizlik rakamları, ortalama günlük yağışlar, kışın çay satışlarında artış, yaz aylarında dondurma satışlarında artış, hepsinde mevsimsel etkiler vardır. Mevsim dalgalanmaları periyodiktir ve uzunlukları hep aynı yani 12 aydır.
- d) *Düzensiz-Tesadüfi hareketler*: Düzensiz ve sistematik olmayan hareketlere (deprem, sel, don, dolu vb.) tesadüfi hareketler denir. İstatiksel yöntemler ancak konjonktür dalgalanmalar ile tesadüfi hareketlerin karışık bir halde belirlenebilmelerine imkan vermektedir. Bu hareketlerin sistematik olmayışı nedeni ile önceden tahmin edilebilmeleri mümkün değildir. Tesadüfi hareketlerin oluşumunda etken olan faktörler doğal ve beklenmeyen olaylardır. Bu hareketlerin herhangi bir yöntemle belirlenmesi mümkün değildir.

2.2. Zaman serilerinde tahmin

Bir zaman serisinin geleceğe yönelik tahmin amacıyla kullanılması serinin sergilediği net hareketlerin gözlenmesi yardımıyla olur. Dolayısıyla, gelecek tahmininde kullanılacak bir serinin, üzerinde etkili olan trend, konjonktür ve mevsim dalgalanmalarının belirlenerek, serinin bu etkilerden arındırılmış olması gerekir.

Zaman serileri deterministik değildir, yani gelecekte ne olacağı kesin olarak tahmin edilemez. Genel olarak, $\{x(t), t = 0, 1, 2, \dots\}$ bir stokastik süreç olan zaman serisinin, $x(t)$ rasgele değişkeninin ortak dağılımını tanımlayan belirli olasılık modelini takip ettiği varsayılmaktadır (Adhikari ve Agrawal 2013). Hipel ve McLeod (1994)'e göre bir zaman serisinin olasılık yapısını tanımlayan matematiksel ifade, stokastik bir süreç

olarak adlandırılır. Dolayısıyla serinin gözlem dizisi aslında onu üreten bir örnek stokastik sürecin gerçekleşmesi olarak tanımlanabilir.

Genel bir varsayım, $x(t)$ zaman serisi değişkenlerinin normal dağılıma ek olarak bağımsız ve aynı şekilde dağılmış olmasıdır. Ancak, Cochrane (1997)'de ele aldığı çalışmasında belirtildiği gibi ilginç olan, zaman serilerinin uzun vadede aslında tamamen bağımsız olmadığı ve aynı şekilde dağılmadıkları, az ya da çok düzenli bir örüntü takip ettikleridir. Örneğin, bugün belirli bir şehrin sıcaklığının aşırı derecede yüksek olması durumunda, yarının sıcaklığının da yüksek olacağı makul bir şekilde kabul edilebilir. Bu, uygun bir teknik kullanarak zaman serileri tahmininin gerçek değere yakın sonuç vermesinin nedenidir.

Zaman serilerinin üç temel özelliği vardır ve bunlar aşağıdaki şekilde tanımlanabilir:

- a) *Dört temel bileşenden meydana gelme özelliği:* Zaman serilerinde meydana gelen dalgalanmalar; ekonomik, politik, mevsimsel gibi çeşitli etkilerin yön ve şiddetinden gelir. Dolayısıyla bu dalgalanmalar trend, konjonktür dalgalanmalar, mevsim dalgalanmaları ve tesadüfi dalgalanmalar olarak dört temel grupta toplanır.
- b) *Bağımlılık Özelliği:* Zaman serilerinin bir diğer özelliği, gözlem değerleri olarak adlandırılan verilerin birbirine bağımlı olmasıdır. Bu özellik sayesinde, bir zaman serisinin bugünkü ve geçmiş dönem gözlem değerleri kullanılarak gelecek dönemde alacağı değerleri tahmin etme imkânı sağlanır.
- c) *Stokastik Süreç Olma Özelliği:* Zaman serileri sadece zamanın deterministik bir fonksiyonu değildir; başka bir deyişle bu olaylar sadece zaman değişkeni tarafından tam olarak açıklanamazlar. Bir zaman serisinin gelecek dönemlerde göstereceği seyri tam olarak açıklayabilmek için kullanılacak matematiksel modelde, bu olayları açıklayacak bütün değişkenlere yer vermek gerekir, ancak bu her zaman mümkün değildir. Modelde bütün değişkenlere yer vermek modeli karmaşıklaştırır ve uygulanabilir olmasını güçleştirir. Ayrıca bütün değişkenler hakkında yeterli bilgi bulunması ve onların sayısal olarak ifade edilmesi mümkün değildir. Zamana bağlı olaylar rassal karakterdedir. Buna deterministik olmayan, stokastik veya istatistik yaklaşım denmektedir. Bu nedenle zaman serileri analiz edilirken bu serilere bir stokastik süreç olarak

bakılması tanımlanması ve analiz için stokastik (ihtimali) modeller kullanılması gereği ortaya çıkmaktadır.

Chatfield (1996)'ya göre uygun bir zaman serisi modeli oluştururken, temel prensibi vardır. Bu ilkeye göre, her zaman mümkün olan en az sayıda parametreye sahip olan ve temel zaman serisi verilerinin yeterli bir gösterimini sağlayan model seçilmelidir. Çünkü model ne kadar karmaşık olursa, gerçek model varsayımlarından ayrılma için o kadar fazla olasılık ortaya çıkacaktır. Model parametrelerinin artmasıyla birlikte, risk fazla takma da daha sonra artar. Dolayısıyla, özet olarak, zaman serileri tahminleri yaparken, diğer tüm olasılıklar arasında en temel modeli seçmek için gerçek bir dikkat gösterilmesi gerektiği söylenebilir.



3. TAHMİN

Bir zaman serisi, gözlemleri bir zaman aralığında belirlenen bir değişken olarak tanımlanır. Bu zaman aralığı saatlik, günlük, haftalık, aylık, mevsimsel, yıllık vb. olabilir. Bu tür veri kümeleri için tahminler yapmak amacıyla son yıllarda araştırmacılar tarafından birçok yöntem araştırılmıştır. Bu yöntemlere zaman serisi tahmin yöntemleri denir.

Zaman serisi tahmin yöntemleri iki kategoride toplanır: olasılıksal tahmin modelleri ve olasılıksal olmayan tahmin modelleri. Geleneksel yöntemler olarak da adlandırılan olasılıksal veya stokastik yöntemler, zaman serilerine bazı varsayımlar getirir. Durağanlık, olasılıklı zaman serisi tahmin yöntemlerinde önemli bir varsayımdır. Bu varsayım, zaman serilerinin sabit ortalama, varyans ve kovaryans fonksiyonuna sahip olmasını gerektirir. Bununla birlikte, uygulamalarda, veri kümelerinin çoğu zaman serilerinin durağanlık veya doğrusallık varsayımını karşılamaz. Bu nedenle, son yıllarda birçok araştırmacı olasılıksal olmayan (alternatif) yöntemler üzerinde çalışır.

3.1 Olasılıksal tahmin modelleri

3.1.1 Otoregresif Entegreli Hareketli Ortalama

Otoregresif Entegreli Hareketli Ortalama (ARIMA) modeli ve farklı varyasyonları Box-Jenkins (1970) tarafından geliştirilir ve bu nedenle bunlar genellikle Box-Jenkins modelleri olarak da bilinir.

ARIMA modelleri, durağan ve durağan olmayan zaman serilerini temsil edebilen bir doğrusal model sınıfıdır. Temeli, yoğun olarak verilerdeki otomatik korelasyon modellerine dayanmaktadır ve genel bir model sınıfından olası bir modeli tanımlamak için etkileşimli bir yaklaşım kullanır. Seçilen model daha sonra veriyi doğru tanımlayıp tanımlamadığını görmek için geçmiş veriler üzerinde test edilir.

ARIMA modellerinin temelinde durağan olmayan bir zaman serisi, veri noktalarının sonlu farkları uygulanarak durağanlaştırılır. Gecikmeli polinomları kullanan ARIMA (p, d, q) modelinin matematiksel formülasyonu Denklem (3.1) ve Denklem (3.2)'de verilmiştir:

$$\varphi(L)(1-L)^d y_t = \theta(L)\varepsilon_t \quad (3.1)$$

Diğer bir deyişle;

$$(1 - \sum_{i=1}^p \varphi_i L^i) (1 - L)^d y_t = (1 + \sum_{j=1}^q \theta_j L^j) \varepsilon_t \quad (3.2)$$

Burada, p , d ve q sifira eşit veya daha büyük tam sayılardır ve sırasıyla modelin otoregresif, tümleşik ve hareketli ortalama parçalarının sırasını belirtir. d tamsayısı, fark alma seviyesini kontrol eder. Model ilk fark alma işleminden sonra durağanlaştı ise $d=1$, ikinci fark alma işleminden sonra durağanlaştı ise $d=2$ alınır. Genellikle $d=1$ yeterlidir. Model kendiliğinden durağansa $d=0$ demektir ve bir ARMA (p, q) modeline indirgenir. ARIMA $(0,1,0)$, yani, $y_t = y_{t-1} + \varepsilon_t$ özel bir durumdur ve Random Walk modeli olarak bilinir. Ekonomik ve hisse senedi fiyat serileri gibi sabit olmayan veriler için yaygın olarak kullanılır (Cochrane 1997 ve Zhang 2003).

3.1.2 Üstel Yumuşatma

Üstel Yumuşatma (ES) yöntemi, temelde zaman serilerini yumuşatmak üzerinedir. Bir zaman serisinin deterministik veya stokastik bir eğilimi varsa, bu zaman serisinin problemlerini tahmin etmek için üstel bir yumuşatma yöntemi uygulanır.

Üstel yumuşatma yöntemleri üç yöntem altında ele alınır: Basit Üstel Yumuşatma Yöntemi, Holt'un Üstel Yumuşatma Yöntemi ve Holt-Winter'in Üstel Yumuşatma Yöntemi (Brockwell ve Davis, 2002).

Simple Üstel Yumuşatma (SES) Yöntemi, zaman serileri ortalamalarının civarında olduğunda kullanılır. Zaman serilerinin ağırlıkları Üstel olarak azalır ve en son gözlemler bu yöntemde ağırlıklandırılır. Zaman serilerini tahmin etmek için aşağıda verilen Denklem (3.3) kullanılmıştır.

$$F_{t+1} = \alpha y_t + (1 - \alpha) F_t \quad (3.3)$$

Burada; F_{t+1} , $t + 1$ periyodu için tahmin değerini, y_t , t periyodu için gerçek değeri ve α , düzeltme sabitini ifade eder.

Double Üstel Yumuşatma olarak da adlandırılan Holt Üstel Yumuşatma yöntemi (Denklem (3.4)-(3.6)), zaman serilerinin eğilimi olduğunda kullanılır. Bu durumda, ikinci yumuşatma sabiti trend için hesaba katılır. Her $0 \leq \alpha \leq 1, 0 \leq \beta \leq 1$ için,

$$L_t = \alpha y_t + (1 - \alpha) (L_{t-1} + b_{t-1}) \quad (\text{seviye}) \quad (3.4)$$

$$b_t = \beta (L_t - L_{t-1}) + (1 - \beta) b_{t-1} \quad (\text{trend}) \quad (3.5)$$

$$F_{t+1} = L_t + b_t \quad (\text{tahmin}) \quad (3.6)$$

Triple Üstel Yumuşatma (HES) yöntemi olarak da adlandırılan Holt-Winter'in Üstel Yumuşatma yöntemi, hem trendi hem de mevsimselliği hesaba katar (Denklem (3.7)-(3.10)).

$$L_t = \frac{y_t}{S_{t-s}} + (1 - \alpha) \left(L_{t-1} + T_{t-1} \right) \quad (\text{seviye}) \quad (3.7)$$

$$T_t = \beta (L_t - L_{t-1}) + (1 - \beta) T_{t-1} \quad (\text{trend}) \quad (3.8)$$

$$S_t = \gamma \frac{y_t}{L_t} + (1 - \gamma) S_{t-s} \quad (\text{mevsimsellik}) \quad (3.9)$$

$$F_{t+p} = (L_t + pT_t) S_{t-1+p} \quad (\text{tahmin}) \quad (3.10)$$

Burada, $0 \leq \alpha \leq 1, 0 \leq \beta \leq 1$ ve $0 \leq \gamma \leq 1$ için s mevsimsel döngünün uzunluğudur.

3.2 Olasılıksal olmayan tahmin modelleri

Gerçek dünya zaman serileri genellikle olasılıksal olmayan bir yapı içerir. Bu nedenle tahminlerin temeli olarak kesin istatistiksel yaklaşımların kullanılması tatmin edici sonuçlar vermeyebilir. Bundan yola çıkarak son birkaç yılda araştırmacılar tarafından olasılıksal olmayan zaman serilerinde tahmin yapabilmek amacıyla birçok alternatif önerilir. (Tak 2018).

3.2.1 Tekrarlayan Tip-1 Bulanık Fonksiyonlar

Tip-1 Bulanık Fonksiyonlar (T1FFs) yaklaşımı, Türksen (2008) tarafından bulanık bir çıkarım sistemi olarak tasarlanır. Klasik bulanık çıkarım sistemleri kural tabanlı



olduğundan, Türksen (2008) çalışmasında kural dışı bir sistem ihtiyacı için T1FF yaklaşımı önerir. Bu, klasik bulanık çıkarım sistemlerine göre büyük bir avantajdır çünkü klasik bulanık çıkarım sistemlerinde kuralları tanımlamak için uzman görüşüne ihtiyaç duyarlar.

T1FFs yaklaşımı başlangıçta sınıflandırma ve regresyon problemleri için önerilir. Bu nedenle, T1FFs'nin tahmin algoritması olarak kullanılabilmesi için yeniden tasarlanması gerekir. T1FFs yaklaşımı ilk kez Beyhan ve Alici (2010) tarafından zaman serisine uyarlanır. Daha sonra Aladağ vd. (2014) çalışmalarında T1FFs'yi zaman serisi tahmin problemlerine uyarlar. Beyhan ve Alici (2010) çalışmalarında bir ARX model yapısı kullanır. Bu yapı en iyi modeli seçemez. Başka bir deyişle, yapı en iyi modeli arama yeteneğine sahip değildir. Bu nedenle Aladağ vd. (2014) çalışmalarında en iyi modeli aramak için bulanık zaman serileri fonksiyon yöntemi önerir ve otoregresif modeli algoritmalarına uyarlar. Sonunda Beyhan ve Alici (2010)'dan daha iyi tahmin sonuçları elde edilir. Geleneksel olasılıklı zaman serisi modelleri arasında otoregresif modeller hareketli modellerin yanı sıra en önemli modellerdendir. Aladağ vd. (2014) yaklaşımlarında otoregresif modeli entegre ederek kullanır.

Son olarak Tak (2018) tarafından Tekrarlayan Tip-1 Bulanık Fonksiyonlar (R-T1FFs) yaklaşımı önerilir. Önerilen yöntemde T1FFs'e otoregresif modellere ek olarak hareketli ortalama modeli de uyarlanır. Önerilen yaklaşımda girdiler zaman serisinin gecikmeli değerleri, hata terimlerinin gecikmeli değerleri ve Bulanık C-Ortalama (Fuzzy C-Means (FCM)) kümeleme yönteminden elde edilen üyelik derecesi olarak alınır. Çalışmada hata terimleri, bulanık fonksiyonların kalıntılarından elde edilir. Bulanık fonksiyonların amaç fonksiyonu, türevlenebilir bir fonksiyon olmadığından amaç fonksiyonunu optimize etmek için Parçacık Sürüsü Optimizasyon algoritması (PSO) önerilir. Amaç fonksiyonu en aza indiren katsayıların tahminini ve dolayısıyla Hata Karelerinin Toplamı (SSE) tahminini elde etmek için önerilen yöntem uyarlanır.

Tekrarlayan Tip-1 bulanık fonksiyonlar için ayarlanacak üç parametre vardır; AR(p) için gecikme sayısı, MA(q) için gecikme sayısı ve küme sayısı. Farklı parametre spesifikasyonlarına sahip tekrarlayan tip-1 bulanık fonksiyonlar için modeller çalışmada bir araya getirilmiştir. Yöntemlerin sonuçlarına bakılarak meta-analiz anlamında yöntemlerin bir araya getirilmesine devam edilmiştir. Sonuçlar, hem

RMSE hem de Ortalama Mutlak Yüzde Hatası (MAPE) açısından tekrarlayan tip-1 bulanık fonksiyonların tahmin performanslarını artırmanın mümkün olduğunu göstermektedir. Çalışmada, yöntemlerin kümelenmesi Bezdek vd. (1981) tarafından tanımlanan Bulanık C-Ortalama (FCM) kümeleme tekniği kullanılarak yapılmıştır. Bu literatürde FCM kullanarak meta-analiz kavramındaki yöntemleri bir araya getirmeyi amaçlayan ilk çalışmadır.

Tak (2018) önerdiği R-TIFFs yaklaşımı aşağıdaki adımlarla açıklamıştır:

Adım 1: Veri kümesi iki gruba ayrılır: Test ve eğitim veri setleri.

Adım 2: Model girdileri, veri setinin gecikmeli değişkenleri ve hataları olarak seçilir. Girdiler Fuzzy C-Means kümeleme (FCM) algoritması kullanılarak kümelenir.

Adım 3: Gecikmeli değişkenler, üyelik dereceleri ve üyelik derecelerinin fonksiyonu eğitim veri kümesinde birleştirilir. Böylece girdi matrisi $X^{(i)(j)}$ elde edilir. Girdi matrisinin boyutu $n * p * c * k$ dir. Burada; n : gözlem sayısı, p : parametre sayısı, c : küme sayısı, k : parçacık sayısıdır.

$$Y^{(i)(j)} = x^{(i)(j)} * \beta^{(i)(j)} + e^{(i)(j)} ; i:1,2,\dots,c , j:1,2,\dots,k \quad (3.11)$$

$$X^{(i)(j)} = \begin{bmatrix} \mu_{i1} & x_{11} & \dots & x_{p1} \\ \vdots & \vdots & \ddots & \vdots \\ \mu_{in} & x_{1n} & \dots & x_{pn} \end{bmatrix} ; Y^{(i)(j)} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \quad \beta^{(i)(j)} = \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_n \end{bmatrix}$$

Adım 4: Üyelik dereceleri, Bulanık C-Ortalama (FCM) algoritması kullanılarak eğitim veri seti için elde edilir. Küme merkezleri de eğitim veri seti için elde edilir. Bu küme merkezleri ve aşağıdaki formül kullanılarak test veri setindeki her bir gözlem için de üyelik dereceleri elde edilir.

$$\mu_{test_i} = \frac{d(X_{test}, center_i)^{-1}}{\sum_{j=1}^c d(X_{test}, center_j)^{-1}} \quad (3.12)$$

Adım 5: μ_{test_i} ve X_{test} her küme ve parçacık için birleştirilir. Örneğin, i^{nci}

küme ve j^{nci} parçacık için girdi matrisi;



$$X_{test} = \begin{bmatrix} \mu_{test} & \log \mu & \mu_{test}^2 & Y_{t-1} & Y_{t-2} & \dots & Y_{t-p} & \epsilon_{t-1} & \epsilon_{t-2} \\ 1 & \cdot & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot \\ 1 & \cdot & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \cdot & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot \end{bmatrix}$$

Adım 6: Parçacık sürüsü optimizasyonu algoritması için C_1 ve C_2 parametreleri, parçacık sayısı ve iterasyon sayısı belirlenir. Her bir parçacıktaki konum sayısı $(p + q + 4) * c$ 'dir. Burada; p: AR için gecikme sayısı, q: MA için gecikme sayısı. c: küme sayısı, 4: X_{test} girdi matrisindeki ilk dört sütunu temsil eder.

Adım 7: Her parçacık için başlangıç pozisyonları standart normal dağılımdan rastgele oluşturulur. β katsayıları $Y^{(i)(j)} = x^{(i)(j)} * \beta^{(i)(j)} + e^{(i)(j)}$ regresyon modelindeki olası çözümleri temsil eder.

Örneğin, ilk parçacık için, p = 4, q = 2 ve c = 2 olduğu varsayılarak, pozisyon sayısı Çizelge 3.1'deki gibi $(4 + p + q) * 2 = (4 + 4 + 2) * 2 = 20$ 'dir.

Çizelge 3.1 : İlk parçacık için pozisyonlar.

c	μ_1	$\log(\mu)_1$	μ^2_1	Y_{t-1}	Y_{t-2}	Y_{t-3}	Y_{t-4}	e_{t-1}	e_{t-2}
β_0	β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9

c	μ_2	$\log(\mu)_2$	μ^2_2	Y_{t-1}	Y_{t-2}	Y_{t-3}	Y_{t-4}	e_{t-1}	e_{t-2}
β_{10}	β_{11}	β_{12}	β_{13}	β_{14}	β_{15}	β_{16}	β_{17}	β_{18}	β_{19}

Pozisyonlara karşılık gelen başlangıç hızları *unif* (0,1) dağılımından rastgele üretilir.

Adım 8: Başlangıçtaki kişisel en iyi (pbest) değerler başlangıç pozisyonları olarak atanır. İlk global en iyi (gbest) değer, kondisyon değeri kullanılarak elde edilir.

Adım 9: Her parçacık için, $e_t - q$ değerleri aşağıdaki formüller kullanılarak hesaplanır.

$$\text{Adım 9.1} \quad \hat{Y}_i^{(j)(k)} = X_{i,}^{(j)(k)} \cdot \beta_{i,}^{T(j)(k)} \quad (3.13)$$

$$\text{Adım 9.2} \quad \hat{Y}_i^{*(k)} = \hat{Y}_i^{(j)(k)} \cdot \mu_{i,}^{T(j)} \quad (3.14)$$

$$\text{Adım 9.3} \quad \epsilon_i^k = Y_i - \hat{Y}_i^{*(k)} \quad (3.15)$$

$$\text{Adım 9.4} \quad X_{i+1,p}^{(\cdot)(k)} = e_i^k \quad (3.16)$$

Adım 9.1'de $\hat{Y}_i^{(k)}$ i'nci küme için zaman serisinin öngörülen değerini, $X_{i,}^{(j)(k)}$ girdi matrisini ve $\beta_{i,}^{T(j)(k)}$ PSO üzerinden i'nci küme için elde edilen konumu temsil eder.

Örneğin,

$$X_{1,}^{(1)(1)} = [C \quad \mu \quad \mu^2 \quad Y_{t-1} \quad Y_{t-2} \quad e_{t-1}]_{1 \times 6},$$

$$\beta_{1,}^{(1)(1)} = [P_1 \quad P_2 \quad P_3 \quad P_4 \quad P_5 \quad P_6]_{1 \times 6},$$

$$\hat{Y}_{1,}^{*(k)} = [\cdot]_{1 \times 1}$$

Adım 9.1. (j) küme sayısı kadar tekrarlanır ve c kümeleri için $\hat{Y}_i^{(k)(j)} =$



değerleri elde edilir. i'nci bulanık fonksiyon için $\hat{Y}_i^{*(k)}$ değeri Adım 9.2'deki şekilde hesaplanır.

Örneğin, birinci obje ve k'nci parçacık için;

$$\hat{Y}_1^{*(\cdot)} = \hat{Y}_1^{(\cdot)(k)} \cdot \mu_{1,}^{T(\cdot)} = [\mu_1^1 \quad \dots \quad \mu_1^c]_{1 \times k} \times \begin{bmatrix} \hat{Y}_1^{(1)(k)} \\ \cdot \\ \hat{Y}_1^{(c)(k)} \end{bmatrix}_{cx1} = [\cdot]_{1 \times 1}$$

ϵ_k^i değeri Adım 9.3'te elde edilir. ϵ_k^i 'nin değeri her küme için aynıdır.

Y_i 'den $\hat{Y}_i^{*(k)}$ çıkarılmasıyla ϵ_k^i elde edilir. Denklem aşağıda verilmiştir.

$$\epsilon_k^i = Y_i - \hat{Y}_i^{*(k)}$$



Örneğin;

$$\epsilon_1^1 = Y_1 - \hat{Y}_{*1}^{(1)}$$

Adım 9.4'te, her küme için X veri setine ϵ_k^i değeri atanır.

Örneğin;

$$X_{2,p}^{(\cdot)(1)} = \epsilon_1^1, \text{ p: parametre sayısı olmak üzere.}$$

Adım 10: Küme sürelerinin sayısı kadar Adım 9.1'i tekrarladıktan sonra, Adım 9.2, Adım 9.3 ve Adım 9.4 hesaplanır. Bu adımlar, gözlem sayısı (n) için birer birer gerçekleştirilir.

Adım 11: Adım 9 ve Adım 10, parçacık zamanı sayısı için tekrarlanır.

Adım 12: Eğitim veri seti için elde edilen gbest ve Adım 9'daki denklemler test veri seti için ϵ_t elde etmek için kullanılır. Denklemler aşağıda verilmiştir:

$$\text{Adım 12.1 } \hat{Y}_{test_i}^{(j)} = X_{test_{i,}}^{(j)} \cdot \beta_{test_{i,}}^{T(j)} \quad (3.17)$$

$$\text{Adım 12.2 } \hat{Y}_{test_i}^* = \hat{Y}_{test_i}^{(j)} \cdot \mu_{test_{i,}}^{T(j)} \quad (3.18)$$

$$\text{Adım 12.3 } \epsilon_{test_i} = Y_{test_i} - \hat{Y}_{test_i}^* \quad (3.19)$$

$$\text{Adım 12.4 } X_{test_{i+1,p}}^{(\cdot)} = \epsilon_{test_i} \quad (3.20)$$

$\beta_{test_{i,}}^{T(j)}$ Adım 9, Adım 10 ve Adım 11'de en iyi değerler olarak elde

edilen katsayılarıdır. $\mu_{test_{i,}}^{T(j)}$ Adım 4'te hesaplanan nesnelere üyelik

derecesidir; Y_{test_i} orijinal zaman serisi değerleridir; $\hat{Y}_{test_i}^*$ tahminlerdir.

Adım 12.1 küme sürelerinin sayısı kadar tekrarlanır, daha sonra Adım

12.2, Adım 12.3 ve Adım 12.4 devam eder ve bu adımlar her gözlem için tek tek tekrarlanır.



Adım 13: r_1 ve r_2 standart normal dağılımdan rastgele üretilir ve yeni pozisyonlar ve hızlar aşağıdaki formüller kullanılarak güncellenir.

$$velocity_{id}^{k+1} = velocity_{id}^k + c_1 \cdot r_1^k \cdot (pbest_{id}^k - p_{id}^k) + c_2 \cdot r_2^k \cdot (gbest^k - p_{id}^k) \quad (3.21)$$

$$p_{id}^{k+1} = p_{id}^k + velocity_{id}^{k+1} \quad (3.22)$$

Adım 14: pbest ve gbest değerleri kondisyon değeri kullanılarak güncellenir.

Adım 15: Adım 9, 10, 11, 12, 13 ve 14 tekrarlama sürelerinin sayısı için tekrarlanır.

Adım 16: Son olarak, aşağıdaki denklemleri kullanarak, zaman serisinin gelecekteki değerlerini tahmin etmek için Denklem (3.23)-(3.24)'de yer alan

$gbest$ değeri $(\beta_{test}^{(j)})$ kullanılır.

$$\hat{Y}_{test_i}^{(j)} = X_{test_i}^{(j)} \cdot \beta_{test_i}^{(j)} \quad (3.23)$$

$$\hat{Y}_{test_i}^* = \hat{Y}_{test_i}^{(j)} \cdot \mu_{test_i}^{(j)} \quad (3.24)$$

Sonuç olarak, $\hat{Y}_{test_i}^*$ değerleri tahmin etmek istediğimiz tahminlerdir.

3.2.1.1 Parçacık Sürüsü Optimizasyonu (PSO)

Kennedy ve Eberhart (1995) tarafından tanıtilen Parçacık Sürüsü Optimizasyonu (PSO) en önemli sürü istihbarat paradigmalarından biridir (Kennedy vd. 2001). PSO, parçacıkların global olarak en uygun çözümleri aramasına rehberlik etmek için kuş sürüsü ve balık okulunda sürü davranışını taklit eden basit bir mekanizma kullanır. PSO'nun uygulanması kolay olduğundan, son yıllarda ve gerçek dünyadaki optimizasyon problemlerinin çözümünde görülen birçok başarılı uygulama ile hızla ilerlemiştir (Eberhart ve Si 2001, Li ve Engelbrecht 2007, Krohling ve Coelho 2006, Franken ve Engelbrecht 2005, Ho vd. 2008, Liu vd. 2007).

Kennedy ve Eberhart (1995) algoritmalarında bir sürünün içindeki her parçacığın rastgele bir konuma sahip olduğunu ve bu pozisyondan optimum arayış içinde



olduklarını ortaya koyar. Her parçacık optimum seviyeye hareket ettiğinden, kendi hızlarına da sahiptir. Her parçacık, iterasyonlar arasındaki en iyi pozisyonunu hatırlar ve buna kişisel en iyi (*pbest*) denir. Kişisel en iyi değerlerin en iyisi, global en iyi değer (*gbest*) olarak tanımlanır. Kişisel en iyiyi kullanarak, her bir parçacık için global en iyi değerleri ve hızları Kennedy ve Eberhart (1995) aşağıdaki algoritmayı kullanarak verir.

PSO algoritması yinelemeli bir algoritmadır. Her yinelemede algoritma, her bir parçacığın kişisel en iyi değerini ve tüm parçacıklar arasındaki global en iyi değeri arar.

Her parçacığın değeri ve hızı, aşağıda verilen vektörlerde saklanır.

$$P_i = (p_{i1}, p_{i2}, \dots, p_{iM})$$

$$V_i = (v_{i1}, v_{i2}, \dots, v_{iM})$$

En iyi global değeri elde etmek için, parçacıklara karşılık gelen uygunluk değerleri karşılaştırılır ve maksimum veya minimum uygunluk değerine sahip parçacık, global en iyi değere atanır. Kişisel en iyi değerler (*pbest_i*) ve global en iyi değerler (*gbest*) aşağıda verilen bir vektörde saklanır.

$$pbest_i = (p_{i1}, p_{i2}, \dots, p_{iM})$$

$$gbest = (p_1, p_2, \dots, p_M)$$

Denklem (3.25) parçacıkların hızını elde etmek için kullanılır:

$$velocity_{im}^{k+1} = velocity_{im}^k + c_1 \cdot r_1^k \cdot (pbest_{im}^k - p_{im}^k) + c_2 \cdot r_2^k \cdot (gbest^k - p_{im}^k) \quad (3.25)$$

Burada;

- $m = 1, 2, \dots, m$ boyuttur.
- k ; yineleme sayısıdır
- r_1^k ; eşit dağılımlı rasgele sayılardır
- c_1 ve c_2 ; ivme katsayılarıdır. İvme katsayılarını belirtmek için çeşitli yollar olmasına rağmen, c_1 ve c_2 çoğu zaman araştırmacılar tarafından 2 olarak alınır. İvme katsayılarını belirtmek için önerilen yöntemde c_1 ve $c_2 = 2$ alınır.

Bir parçacığın hızını güncelledikten sonra bir sonraki adım parçacığın konumlarını güncellemektir. Bunun için aşağıda verilen Denklem (3.26) kullanılır.

$$p = p_{im}^k = velocity_{im}^{k+1} \quad (3.26)$$

Bu güncellemeler yineleme süresine kadar tekrarlanır. PSO için algoritma aşağıdaki adımlarla özetlenebilir (Tak vd. 2018) :

Adım 1 : Her parçacığın konumu rastgele başlatılır.

$$X_i = (x_{i1}, x_{i2}, \dots, x_{iM})$$

Adım 2 : Her bir konuma karşılık gelen hızlar rastgele başlatılır.

$$P_i = (p_{i1}, p_{i2}, \dots, p_{iM})$$

Adım 3 : Değerlendirme fonksiyonu kullanılarak, $pbest_i$ ve $gbest$ değerleri güncellenir.

$$pbest_i = (p_{i1}, p_{i2}, \dots, p_{iM})$$

$$gbest = (p_1, p_2, \dots, p_M)$$

Adım 4 : c_1 ve c_2 'nin ikiye eşit olmasına izin verilir.

Adım 5 : Sürüdeki hızlar ve pozisyonlar sırasıyla (3.25) ve (3.26) 'de verilen denklemlerle güncellenir.

3.2.2 Uyarlanabilir Nöro-Bulanık Çıkarım Sistemleri

Uyarlanabilir Nöro-Bulanık Çıkarım Sistemleri (ANFIS), Jang (1993) tarafından önerilmiştir. En yaygın kullanılan ve üzerinde çalışılan algoritmalarından biri olduğu için, bulanık öğrenme toplumu için çok önemli bir araçtır. Çelikyılmaz ve Türksen (2009)'a göre, "ANFIS, sinir ağlarının öğrenme yeteneklerini bulanık çıkarım sistemlerine getiren nöro-bulanık bir tekniktir."

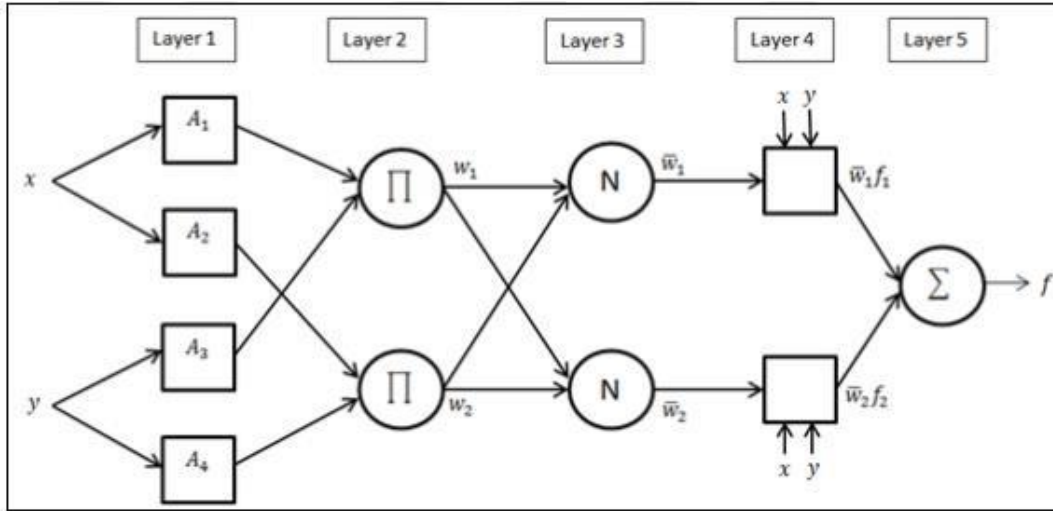
Uyarlanabilir bir ağ, düğümlerden ve yönlü bağlantılardan oluşur. İki x_1 ve x_2 girdisi ve bir çıktı değişkeni y ve iki bulanık kuralı olan bulanık bir sisteme sahip olduğumuzu

varsayalım. Bu bulanık sistemin grafik temsili Şekil 3.1'de verilmiştir. ANFIS'in en çok kullanılan yapısı, aşağıdaki gibi tanımlanan Sugeno tipi bulanık çıkarım sistemidir: (Tak, 2016)

Kural 1: IF x is A_1 AND y is A_3 THEN $f_1 = p_1x + q_1 y + r_1$

Kural 2: IF x is A_2 AND y is A_4 THEN $f_2 = p_2x + q_2 y + r_2$

Şekil 3.1'den de görebileceğimiz gibi, ANFIS 5 katmandan oluşmaktadır.



Şekil 3.1 : İki girdili 2 kurallı ANFIS.

Katman 1 (Bulanıklaştırma) : Bu katmanlardaki girdileri dilbilimsel terimlere dönüştürmek için Bulanık-C Ortalama (FCM) algoritması kullanılır.

Katman 2 (Öncüllerin Toplanması) : Ürün katmanı olarak da adlandırılır, bu katmandaki her düğüm sabit bir düğümdür. Çıkış, tüm gelen sinyallerin ürünüdür. Katmanın çıktısı, Denklem (3.27)'deki gibi resmileştirilebilir.

$$w_i = \mu_{A_i}(x) * \mu_{B_i}(y) \quad ; \quad i=1,2 \quad (3.27)$$

Bu katmanda her düğüm, kuralın ateşleme gücünü temsil eder.

Katman 3 (Ateşleme Derecelerinin Normalleştirilmesi) : Bu katmandaki her düğüm sabit bir düğümdür. i . Düğüm, i . Cetvelin ateşleme gücünün tüm ruletin ateşleme

kuvvetlerinin toplamına oranını hesaplar. Çıktılara normalize ateşleme gücü denir ve Denklem (3.28)'deki şekilde elde edilir.

$$\bar{w} = \frac{w_i}{w_2 + w_1}, i = 1,2 \quad (3.28)$$

Katman 4 (Uygulama) : Bu katmandaki her düğüm, Denklem (3.29)'te verilen düğüm işlevine sahip uyarlanabilir bir düğümdür.

$$\bar{w}_i f_i = \bar{w}_i (p_i x + q_i y + r_i) \quad (3.29)$$

\bar{w}_i , Katman 3'ten normalleştirilmiş ateşleme kuvveti olduğunda, $\{p_i + q_i + r_i\}$ bu düğümün parametre setidir. Bunlara sonradan gelen parametreler denir.

Katman 5 (Sonuçların Birleştirilmesi) : Bu katmandaki tek düğüm, Denklem (3.30)'te verilen genel çıktıyı hesaplayan sabit bir düğüm etiketli toplamdır.

$$\sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum w} \quad (3.30)$$

3.2.3 Yapay Sinir Ağları

Yapay Sinir Ağları (ANN), sosyal, ekonomik, mühendislik, döviz, hisse senedi problemleri vb. Tahminlerinde verimli uygulamalardan yararlanan en doğru ve yaygın olarak kullanılan tahmin modellerinden biridir. ANN'nin çeşitli ayırt edici özellikleri onları değerli ve çekici kılar. Birincisi, geleneksel model tabanlı yöntemlerin aksine, ANN, incelenen problemler için modeller hakkında birkaç ön varsayıma sahip olduğu için veriye dayalı kendi kendine uyarlanabilir yöntemlerdir. İkincisi, ANN genelleştirebilir. ANN'ler kendilerine sunulan verileri (bir örneklem) öğrendikten sonra, örnek veriler gürültülü bilgiler içerse bile, popülasyonun görünmeyen kısmını doğru bir şekilde çıkarabilir. Üçüncüsü, ANN'ler evrensel işlevsel tahmin edicilerdir. Bir ağın herhangi bir sürekli işlevi istenen herhangi bir doğruluğa yaklaştırabileceği gösterilmiştir. Son olarak, ANN doğrusal değildir. Box – Jenkins veya ARIMA gibi zaman serisi tahminine yönelik geleneksel yaklaşımlar, incelenen zaman serilerinin doğrusal süreçlerden üretildiğini varsayar. Ancak, altta yatan mekanizma doğrusal değilse uygun olmayabilir (Khashei ve Bijari 2010).

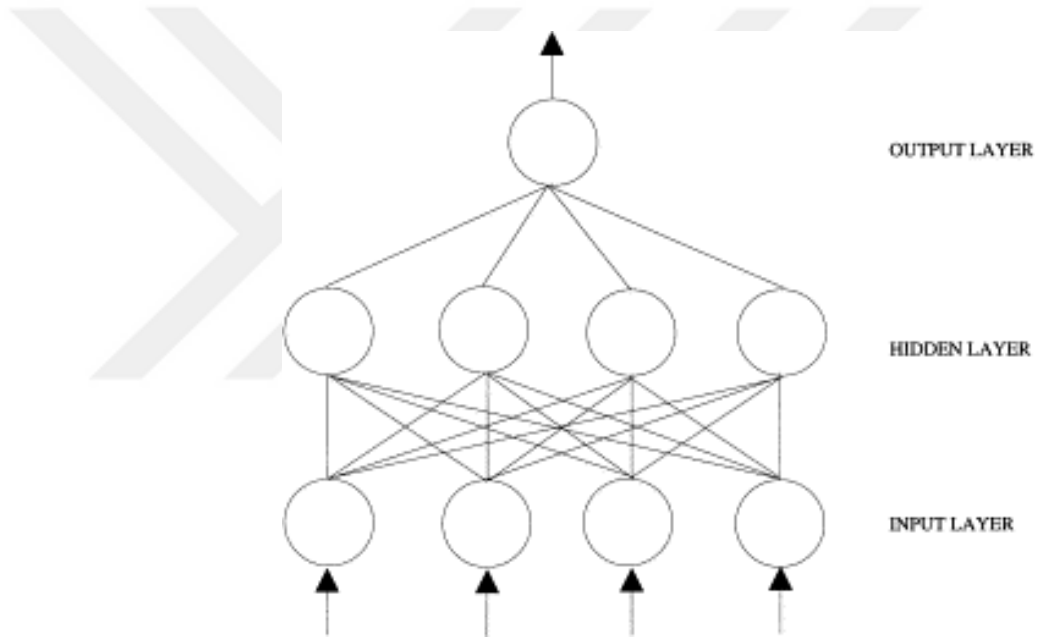
ANN yaklaşımı, zaman serisi tahmini için yararlı bir araç ve iyi bilinen bir yöntemdir (Yolcu vd. 2013). Ancak tahmin için ANN kullanma fikri yeni değildir. İlk olarak Hu (1964) çalışmasında Widrow'un hava tahminine uyarlanabilir doğrusal ağını kullanır. O sırada genel çok katmanlı ağlar için bir eğitim algoritmasının olmaması nedeniyle araştırma oldukça sınırlıdır. 1986'da geri yayılım algoritmasının tanıtıldığı zamana kadar (Rumelhart ve ark (1986b), Werbos, (1988)), tahmin için ANN kullanımında çok fazla gelişme olur. Werbos (1974), Werbos (1988) ilk olarak geri yayılımı (back propagation) formüle eder ve geri yayılımla eğitilen ANN'lerin regresyon ve Box-Jenkins yaklaşımları gibi geleneksel istatistiksel yöntemlerden daha iyi performans gösterdiğini bulur. Lapedes ve Farber (1987), simüle edilmiş bir çalışma yürütür ve ANN'lerin doğrusal olmayan zaman serilerini modellemek ve tahmin etmek için kullanılabileceği sonucuna varır. Weigend vd. (1990), Weigend vd. (1992), Cottrell vd. (1995), gerçek dünya zaman serilerini tahmin etmek için ağ yapısı konusunu ele alır. Tang vd. (1991), Sharda ve Patil (1992) ve Tang ve Fishwick (1993), diğerlerinin yanı sıra, Box-Jenkins ve ANN modelleri arasındaki birkaç tahmin karşılaştırmasının sonuçlarını rapor etmektedir.

Başlangıçta temel biyolojik sinir sistemlerini taklit etmek için geliştirilen ANN özellikle insan beyni, nöronlar veya düğümler adı verilen bir dizi birbirine bağlı basit işlem ögesinden oluşur. Her bir düğüm, diğer düğümlerden veya harici uyarılardan gelen toplam "bilgi" olan bir giriş sinyali alır, bunu bir aktivasyon veya transfer fonksiyonu aracılığıyla yerel olarak işler ve diğer düğümlere veya harici çıkışlara dönüştürülmüş bir çıkış sinyali üretir. Her bir nöron işlevini oldukça yavaş ve kusurlu bir şekilde yerine getirirse de, toplu olarak bir ağ, şaşırtıcı sayıda görevi oldukça verimli bir şekilde gerçekleştirebilir (Reilly ve Cooper, 1990). Bu bilgi işleme özelliği, ANN'leri güçlü bir hesaplama cihazı yapar ve örneklerden öğrenebilir ve daha sonra daha önce hiç görülmemiş örneklere genelleme yapabilir.

1980'lerden beri birçok farklı ANN modeli önerilir. Belki de en etkili modeller, çok katmanlı algılayıcılar, Hopfield ağları ve Kohonen'in kendi kendini organize eden ağlarıdır. Hopfield (1982), ilişki bellemek için çalışan tekrarlayan bir sinir ağını önermektedir. İlişkilendirilebilir bir bellek, kısmi veya bozuk bir versiyondan bir örneği hatırlayabilir. Hopfield ağları, düğümler arasında tam bir ara bağlantı ile katmanlı değildir. Ağın çıktıları, mutlaka girdilerin işlevleri değildir. Daha ziyade,

yinelemeli bir sürecin kararlı durumlarıdır. Kohonen'in özellik haritaları (Kohonen, 1982), insan beyninin kendi kendini organize etme davranışı tarafından motive edilir.

Bir çok katmanlı algılayıcı tipik olarak birkaç düğüm katmanından oluşur. İlk veya en alt katman, harici bilgilerin alındığı bir giriş katmanıdır. Son veya en yüksek katman, problem çözümünün elde edildiği bir çıktı katmanıdır. Giriş katmanı ve çıktı katmanı, gizli katmanlar adı verilen bir veya daha fazla ara katmanla ayrılır. Bitişik katmanlardaki düğümler genellikle daha düşük bir katmandan daha yüksek bir katmana tamamen bağlanır. Şekil 3.2, tek bir gizli katmana sahip tamamen bağlı bir çok katmanlı algılayıcı örneğini vermektedir.



Şekil 3.2 : Tipik ileri beslemeli çok katmanlı algılayıcılar.

Zhang vd. (1998) çalışmalarında ANN'yi detaylı olarak incelemiştir. Zhang vd. (1998)'ye göre ANN metodolojisi aşağıdaki gibidir:

- Açıklayıcı veya nedensel bir tahmin problemi için, bir ANN'nin girdileri genellikle bağımsız veya öngörücü değişkenlerdir. ANN tarafından tahmin edilen fonksiyonel ilişki şu şekilde yazılabilir:

$$y = f(x_1, x_2, \dots, x_p) \quad (3.31)$$

Burada x_1, x_2, \dots, x_p ve p bağımsız değişkenlerdir ve y bir bağımlı değişkendir. Bu anlamda, sinir ağı işlevsel olarak doğrusal olmayan bir regresyon modeline eşdeğerdir.

- Öte yandan, bir ekstrapolatif veya zaman serisi tahmin problemi için girdiler tipik olarak veri serilerinin geçmiş gözlemleridir ve çıktı, gelecekteki değerdir. Bu nedenle ANN aşağıdaki işlev eşlemesini gerçekleştirir.

$$y_{t+1} = f(y_t, y_{t-1}, \dots, y_{t-p}) \quad (3.32)$$

Burada y_t , t zamanındaki gözlemdir. Bu nedenle ANN, zaman serisi tahmin problemleri için doğrusal olmayan otoregresif modele eşdeğerdir. Aynı zamanda, hem tahmin edici değişkenleri hem de zaman gecikmeli gözlemleri, genel transfer fonksiyonu modeli anlamına gelen tek bir ANN modeline dahil etmek kolaydır.

Bir ANN, istenen herhangi bir görevi gerçekleştirmek için kullanılmadan önce, bunu yapmak için eğitilmelidir. Temel olarak eğitim, ANN'nin temel unsurları olan yay ağırlıklarını belirleme sürecidir. Bir ağ tarafından öğrenilen bilgi, yay ağırlıkları ve düğüm önyargıları şeklinde yaylarda ve düğümlerde depolanır. Bir ANN, giriş düğümlerinden çıkış düğümlerine karmaşık doğrusal olmayan eşleştirmeleri gerçekleştirebilir. Bir çok katmanlı algılayıcı eğitimi, her bir giriş model için ağı istenen yanıtının (hedef değer) her zaman mevcut olması bakımından denetlenen bir eğitimidir.

Eğitimin girdi verileri, girdi değişkenlerinin vektörleri veya eğitim modelleri biçimindedir. Bir giriş vektöründeki her bir elemana karşılık gelen, ağ giriş katmanındaki bir giriş düğümüdür. Dolayısıyla, giriş düğümlerinin sayısı, giriş vektörlerinin boyutuna eşittir. Nedensel bir tahmin problemi için, giriş düğümlerinin sayısı iyi tanımlanır ve problemle ilişkili bağımsız değişkenlerin sayısıdır. Bununla birlikte, bir zaman serisi tahmin problemi için, uygun sayıda giriş düğümünün belirlenmesi kolay değildir.

Boyut ne olursa olsun, bir zaman serisi tahmin problemi için girdi vektörü neredeyse her zaman seri boyunca sabit uzunlukta hareketli bir pencereden oluşur. Mevcut toplam veriler genellikle bir eğitim setine (örnek içi veriler) ve bir test setine (örneklem

dışı veya kalıcı örnek) bölünür. Eğitim seti, ağın genelleme yeteneğini ölçmek için test seti kullanılırken, yay ağırlıklarının tahmin edilmesi için kullanılır.

Eğitim süreci genellikle aşağıdaki gibidir:

- İlk olarak, eğitim setinin örnekleri giriş düğümlerine girilir. Giriş düğümlerinin aktivasyon değerleri ağırlıklandırılır ve birinci gizli katmandaki her düğümde toplanır.
- Toplam, daha sonra bir aktivasyon fonksiyonu tarafından düğümün aktivasyon değerine dönüştürülür. Sonuçta çıktı aktivasyon değerleri bulunana kadar bir sonraki katmandaki düğümlere bir girdi haline gelir.
- Eğitim algoritması, karesel hataların toplamı veya ortalama kare hataları gibi bazı genel hata ölçümlerini en aza indiren ağırlıkları bulmak için kullanılır. Dolayısıyla, ağ eğitimi aslında sınırlandırılmamış doğrusal olmayan bir minimizasyon problemidir.

Bir zaman serisi tahmin problemi için, bir eğitim modeli, serinin sabit sayıda gecikmeli gözleminden oluşur. Eğitim setinde N adet y_1, y_2, \dots, y_n gözlemimiz olduğunu ve 1 adım ileri tahmine ihtiyacımız olduğunu, sonra n giriş düğümlü ANN kullandığımızı varsayalım, $N - n$ eğitim modelimiz vardır. İlk eğitim modeli girdi olarak y_1, y_2, \dots, y_n ve hedef çıktı olarak y_{n+1} 'den oluşacaktır. İkinci eğitim modeli, girişler olarak y_2, y_3, \dots, y_{n+1} ve istenen çıktı olarak y_{n+2} içerecektir. Son olarak, son alıştırma modeli, girişler için $y_{N-n}, y_{N-n+1}, \dots, y_{N-1}$ ve hedef için y_N olacaktır.

Tipik olarak, eğitim süreci sırasında en aza indirilecek bir hata karelerinin toplamı tabanlı amaç fonksiyonu veya maliyet fonksiyonu, a_i , ağın gerçek çıktısı olmak üzere şu şekildedir:

$$E = (n + 1)N(y_i, a_i)^2 \quad (3.33)$$

Literatürde doğrusal zaman serileri için ARIMA gibi doğrusal modeller kullanılırken, ANN doğrusal olmayan zaman serileri için tercih edilir. Gerçek hayattaki zaman serilerinin genellikle hem trend hem de mevsimsel varyasyonları içerebileceği bilinen bir gerçektir.

Bir zaman serisinin saf doğrusal veya saf doğrusal olmayan olması neredeyse imkansızdır. Bazı zaman serileri için, doğrusal modeller, zaman serisinin doğrusal bölümü doğrusal olmayan bölümden üstün olduğunda tatmin edici sonuçlar üretebilir. Benzer şekilde, doğrusal olmayan bölüm zaman serisinin doğrusal kısmına göre, doğrusal olmayan modeller tatmin edici sonuçlar verebilir. Ancak her iki durumda da bu parçalardan biri dikkate alınmaz. Böylece aldatıcı sonuçlara yol açabilir.

Bu problemin üstesinden gelmek için literatürde çeşitli hibrit yaklaşımlar önerilmiştir. Tseng vd. (2002), mevsimsel ARIMA (SARIMA) ve ANN'yı birleştiren karma bir tahmin modeli önerir. Zhang (2003), ARIMA ve ANN'ye dayalı bir hibrit model geliştirir. Zhang (2003), yönteminde, zaman serilerinin doğrusal ve doğrusal olmayan parçaların toplamından oluştuğu varsayılır. Zhang (2003), tarafından önerilen yöntemde, doğrusal kısım ARIMA tarafından analiz edilirken, ARIMA'dan elde edilen kalıntılar, servis için besleme ANN ile modellenir. BuHamra vd. (2003) ve Jain ve Kumar (2007), ANN girdilerinin Box-Jenkins (1970) prosedürü ile belirlendiği hibrit yaklaşımları önerir. Bu çalışmalara ek olarak, SARIMA ve ANN'yi birleştiren hibrit yaklaşımların da bulanık zaman serilerini analiz etmesi önerilir (Eğrioğlu vd. 2009; Uslu vd. 2010)

3.2.4 Uzun Kısa Süreli Bellek Ağları

Uzun Kısa Süreli Bellek Ağları (LSTM), uzun vadeli bağımlılıkları öğrenebilen özel bir tür tekrarlayan sinir ağlarıdır. Hochreiter ve Schmidhuber (1997) tarafından geliştirilen LSTM ağları, pek çok kişi tarafından takip eden çalışmalarda kullanılır. Çok çeşitli problemler üzerinde çok iyi sonuç vermekte olup günümüzde yaygın olarak kullanılmaktadır. LSTM ve geleneksel tekrarlayan sinir ağları çeşitli dizi tahmini ve dizi etiketleme görevlerine başarıyla uygulanır (Sak ve ar. 2014).

LSTM, uzun vadeli bağımlılık sorununu önlemek için açıkça tasarlanır. Bilgiyi uzun süre hatırlamak, pratik olarak onların varsayılan davranışdır, öğrenmekte zorlandıkları bir şey değildir.

LSTM derin öğrenme algoritmalarındandır ve değişkenler arasındaki ilişkilerin derin ve katmanlı bir hiyerarşi içinde modellendiği tahmin problemlerine yeni yaklaşımlar getirir. LSTM gibi derin öğrenmeye dayalı algoritmalar, finans dahil birçok disiplinde uygulamaları ile son zamanlarda ilgi görmektedir.

Derin öğrenme yöntemleri, zaman serisi tahmininde doğrusal olmama ve karmaşıklık gibi verilerin yapısını ve modelini belirleyebilir. Özellikle, LSTM, Brownlee (2016), Gers vd. (2000), Hochreiter (1997) çalışmalarında zaman serisi tahmini için kullanılır.

Namini vd.(2018)'e göre ilginç ve önemli bir araştırma sorusu, derin öğrenme tabanlı tahmin algoritmalarıyla karşılaştırıldığında geleneksel tahmin tekniklerinin doğruluğu ve kesinliğidir. Bildiğimiz kadarıyla, performansını değerlendirmek ve geleneksel ARIMA gibi tahmin yöntemleri ile ekonometrik olarak karşılaştırmak için ekonomik ve finansal zamanlayıcı serisi verilerini tahmin etmede LSTM yönteminin üstünlüğünü ifade eden belirli bir ampirik kanıt yoktur. Namini vd.(2018) çalışmasında, ARIMA ve LSTM modellerini Kök Ortalama Kare Hatası (RMSE) oranlarını azaltmadaki performansları açısından karşılaştırmaktadır. Geleneksel tahmin modellemesinin bir temsilcisi olarak ARIMA, toplanan ve modellenen verilerin durağan olmayan özelliği nedeniyle seçilir. Benzer bir şekilde ve derin öğrenme tabanlı algoritmaların bir temsilcisi olarak, LSTM yöntemi verilen verilerin özelliklerinin daha uzun süre korunması ve eğitilmesinde kullanılması nedeniyle seçilir. Çalışmada, bir dizi ekonomik ve finansal zaman serisi verisi için veri işleme ve LSTM modellerinin eğitimi hakkında derinlemesine bir rehberlik sunulur.

Qing ve Niu (2018) çalışmasında Güneş ışınımının tahmin problemini, aynı anda birden fazla çıktıyı birlikte tahmin eden yapısal çıktı tahmin problemi olarak formüle etmektedir. Önerilen tahmin modeli, aynı günün birbirini izleyen saatleri arasındaki bağımlılığı hesaba katan LSTM ağları kullanılarak eğitilir. Güneş ışınım tahmini için geri yayılım algoritması kullanarak kalıcılık algoritmasını, doğrusal en küçük kare regresyonunu ve çok katmanlı ileri beslemeli sinir ağlarını karşılaştırılır. Cape Verde'deki Santiago adasında toplanan bir veri kümesindeki deneysel sonuçlar, önerilen algoritmanın tek çıktı tahmini için bu rekabetçi algoritmalarından daha iyi performans gösterdiğini göstermektedir. Önerilen algoritma, yarı yıllık test verilerini tahmin etmek için yaklaşık 2 yıllık eğitim verilerini kullanarak, RMSE açısından geri yayılım algoritmasından % 18.34 daha doğrudur. 1 yıllık ışınım verisini tahmin etmek için 10 yıllık geçmiş verileri kullanan bir durum için, önerilen LSTM algoritmasını kullanan tahmin RMSE, geri yayılım algoritmasına göre% 42,9 azalır.

LSTM'ye girmeden önce, bir tekrarlayan sinir ağının neye benzediğine bir göz atmak gerekir. Tekrarlayan Sinir Ağı (RNN), amacın dizide gözlemlenen önceki adımlara

göre gözlemler dizisindeki bir sonraki adımı tahmin etmek olduğu özel bir sinir ağı durumudur. Aslında, RNN'lerin arkasındaki fikir, sıralı gözlemlerden yararlanmak ve gelecekteki eğilimleri tahmin etmek için önceki aşamalardan öğrenmektir. Sonuç olarak, sonraki adımlar tahmin edilirken önceki aşamalar verilerinin hatırlanması gerekir.

RNN'lerde, gizli katmanlar, sıralı verileri okumanın önceki aşamalarında yakalanan bilgileri depolamak için dahili depolama görevi görür. RNN'ler "tekrarlayan" olarak adlandırılır çünkü sıranın her ögesi için aynı görevi yerine getirirler ve daha önce yakalanan bilgileri gelecekteki görünmeyen sıralı verileri tahmin etmek için kullanma özelliği ile birlikte. Tipik bir genel RNN ile ilgili en büyük zorluk, bu ağların dizideki yalnızca birkaç önceki adımı hatırlaması ve bu nedenle daha uzun veri dizilerini hatırlamaya uygun olmamasıdır. Bu zorlu sorun, LSTM'in tekrarlayan ağda sunulan "Bellek Kapısı" kullanılarak çözülür.

LSTM, veri sırasını ezberlemek için ek özelliklere sahip özel bir RNN türüdür. Her LSTM, veri akışlarının yakalandığı ve depolandığı bir dizi hücre veya sistem modülleridir. Hücreler, bir modülden diğerine bağlanan ve geçmişten gelen verileri taşıyan ve bunları şimdiki için toplayan bir taşıma hattına (her hücredeki üst çizgi) benzer. Her hücrede bazı kapıların kullanılması nedeniyle, her hücredeki veriler sonraki hücrelere atılabilir, filtrelenebilir veya eklenebilir. Böylelikle, sigmoidal sinir ağı katmanına dayanan kapılar, hücrelerin isteğe bağlı olarak verilerin geçmesine veya atılmasına izin vermesini sağlar.

Her sigmoid katmanı, her hücrede izin verilmesi gereken her veri segmentinin miktarını gösteren, sıfır ve bir aralığında sayılar verir. Daha doğrusu, sıfır değer tahmini, "hiçbir şeyin geçmesine izin vermeyin" anlamına gelir; buna karşılık; "1" tahmini, "her şeyin geçmesine izin ver" anlamına gelir. Her bir hücrenin durumunu kontrol etmek amacıyla her LSTM'de üç tür kapı bulunur:

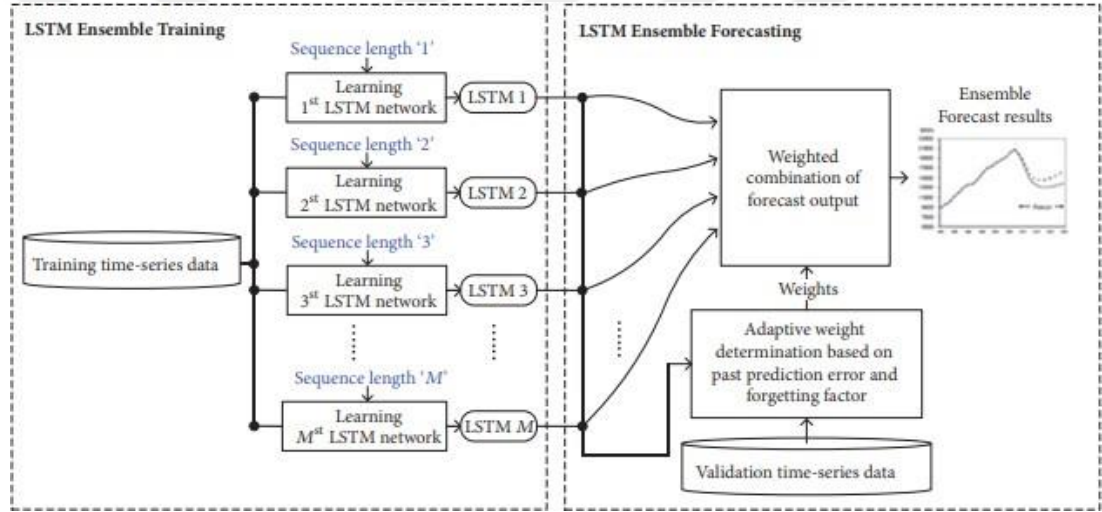
- *Unut Kapısı*: 0 ile 1 arasında bir sayı verir, burada 1 "bunu tamamen koru" gösterir; oysa 0, "bunu tamamen göz ardı et" anlamına gelir.
- *Bellek Kapısı*: Hücrede hangi yeni verilerin depolanması gerektiğini seçer. İlk olarak, "giriş kapısı katmanı" adı verilen bir sigmoid katman, hangi değerlerin

değiştirileceğini seçer. Daha sonra, bir tanh katmanı, duruma eklenebilecek yeni aday değerlerin bir vektörünü oluşturur.

- *Çıktı Kapısı*: Her hücreden ne verileceğine karar verir. Verilen değer, filtrelenmiş ve yeni eklenen verilerle birlikte hücre durumuna bağlı olacaktır.

LSTM akış şeması Şekil 3.3'te verilmiştir Choi ve Lee (2018).

Regresyon kullanan modellemenin aksine, zaman serisi veri setlerinde girdi değişkenleri arasında bir bağımlılık dizisi vardır. Tekrarlayan Sinir Ağları, girdi değişkenleri arasındaki bağımlılığı ele almakta çok güçlüdür. LSTM, uzun gözlem dizisini tutabilen ve bunlardan öğrenebilen bir RNN türüdür. Geliştirilen algoritma, çok adımlı tek değişkenli bir tahmin algoritmasıdır. (Brownlee 2016).



Şekil 3.3 : LSTM akış şeması.

Bir LSTM ağı, bir $x = (x_1, x_2, \dots, x_t)$ giriş dizisinden $y = (y_1, y_2, \dots, y_t)$ çıkış dizisine olan eşlemeyi, $t = 1$ 'den T 'ye olmak üzere aşağıdaki Denklem (3.34)-(3.39) için yinelemeli olarak kullanarak ağ birimi aktivasyonlarını elde ederek hesaplar (Sak vd. 2014, Choi ve Lee 2018):

$$\dot{i}_t = \sigma (W_{ix}x_t + W_{im}m_{t-1} + W_{ic}c_{t-1} + b_i) \quad (3.34)$$

$$f_t = \sigma (W_{fx}x_t + W_{fm}m_{t-1} + W_{fc}c_{t-1} + b_f) \quad (3.35)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g(W_{cx}x_t + W_{cm}m_{t-1} + b_c) \quad (3.36)$$

$$o_t = \sigma(W_{ox}x_t + W_{om}m_{t-1} + W_{oc}c_t + b_o) \quad (3.37)$$

$$m_t = o_t \odot h(c_t) \quad (3.38)$$

$$y_t = \phi(W_y m_t + b_y) \quad (3.39)$$

Burada, W terimleri ağırlık matrislerini belirtir (örneğin, W_{ix} , giriş kapısından girişe kadar ağırlık matrisidir), W_{ic} , W_{fc} , W_{oc} gözetleme deliği bağlantıları için köşegen ağırlık matrisleridir, b terimleri sapma vektörlerini belirtir (b_i , giriş kapısıdır bias vektörü), σ lojistik sigmoid fonksiyonudur ve i , f , o ve c sırasıyla giriş kapısı, unutma kapısı, çıkış kapısı ve hücre aktivasyon vektörleridir, bunların tümü hücre çıkış aktivasyon vektörü m ile aynı boyuttadır, vektörlerin element-bilge çarpımıdır, g ve h genel olarak hücre girişi ve hücre çıkışı aktivasyon fonksiyonlarıdır ve ϕ softmax ağ çıkış aktivasyon fonksiyonudur.

3.2.5 Meta Bulanık Fonksiyonlar

Meta analizin temel amacı, belirli bir konuda çok sayıda bilimsel çalışmanın sonuçlarını bir araya getirmektir. Meta analiz Glass (1976) tarafından tanıtılır. Makalesinde 375 farklı psikoterapi sonuç çalışmasının bulgularını istatistiksel olarak bir araya getirir. DerSimonian ve Laird (1986), ele aldıkları çalışmada meta analizin, analitik sonuçların bir derlemesi olarak bulguları bütünleştiren bir çalışma olduğunu tanımlar. Meta analize dayalı çalışmalar, son birkaç yılda daha popüler hale gelir. Tak (2018), çalışmasında bir amaç için farklı çalışmaların sonuçlarını bir araya getirmek yerine, bir amaca yönelik farklı yöntemleri bir araya getirir. Farklı yöntemlerin toplanmasının veya hibritlenmesinin daha iyi tahmin veya tahmin doğruluğuna sahip olduğunu gösterir. Bu nedenle, makalenin ana katkısı, işlevlerde toplayabildiğimiz kadar çok yöntemi bir araya getirmek olmuştur.

"Veri kümesi için hangi yöntemi seçmeliyiz?" ve "Lütfen önerilen yöntemin diğer yöntemlerden neden daha iyi performans göstermediğini açıklayın?" gibi sorular bizi Meta Bulanık Fonksiyonlar (MFFs)'leri bulmaya yönlendirir. MFFs'nin ana kapsamı, işlevler halinde toplayabildiğimiz kadar çok yöntemi toplamak ve sonunda daha iyi

sonuçlar elde etmektir. Bu nedenle, MFFs'nin girdi matrisi, daha önce tanıtilen yöntemlerin çıktılarında oluşur. Yöntemlerin çıktıları karakteristik olarak kullanılarak, yöntemler FCM kullanılarak kümelendirir. Son olarak, en iyi küme (işlev), MFFs'nin en iyisi olarak seçilir Tak (2018). MFFs'lerin kurulumunda başka bir bulanık kümeleme tekniğinin kullanılması gelecekteki çalışma olabilir.

MFFs algoritmasında açıklığa kavuşturulması gereken üç bileşen vardır. Birincisi, "Fonksiyon" un netleştirilmesidir. Fonksiyon, yöntemlerin birleşiminden oluşur. "Fonksiyon" dediğimizde, FCM'de "küme" demek istiyoruz. İkincisi, fonksiyonlardaki yöntemlerin ağırlıklarıdır. Yöntemlerin ağırlıkları, basitçe bir kümedeki bir yöntemin üyelik derecelerinden elde edilir. Üçüncüsü, en iyi meta fuzzyfonksiyon " MFF_{best} ". Küme sayısı kadar fonksiyona sahip olduğumuz için, en iyi değerlendirme kriterlerine sahip bir fonksiyon arıyoruz. Bu durumda, en iyi değerlendirme kriterlerine sahip olan işleve MFF_{best} adı verilir ve gelecekteki tahminler veya tahminler MFF_{best} ile hesaplanır.

MFFs'nin amacı, aynı amaca yönelik yöntemleri fonksiyonlar halinde birleştirmektir. Yöntemleri bir araya getirmenin arkasındaki fikir, her yöntemin belirli bir veri kümesi için çok fazla veya kısmi bilgiye sahip olduğu veya hiç bilgisi olmadığı varsayımıdır. Böylece, daha iyi performans gösteren yöntemler tek bir işlevde toplanırken, daha kötü performans gösteren yöntemler başka bir işlevde toplanacaktır. Amacımız, birçok yöntemin gücünü kullanarak daha iyi sonuçlar elde etmek veya en azından birçoğunu arasından en iyi yöntemi elde etmektir.

MFFs algoritması aşağıdaki şekilde tanımlanır:

Adım 1: Problem için mevcut tahmin yöntemlerini belirleyin.

Adım 2: X veri seti, train (X_{train}) ve test (X_{test}) olmak üzere iki gruba ayrılır.

$$X = [X_{ij}] , i = 1,2 \dots, p ; j = 1,2, \dots, n$$

$$X_{train} = [X_{ij}] ; i = 1,2 \dots, p ; j = 1,2, \dots, n_{train}$$

$$X_{test} = [X_{ij}] ; i = 1,2 \dots, p ; j = n_{train+1} , n_{train+2} , \dots, n$$

Adım 3: X_{train} kullanılarak ilgili veri kümesi için ilgili mevcut yöntemler eğitilir.

Adım 4: Test veri kümesi X_{test} için eğitilmiş yöntemler kullanılarak sonuçlar (yani tahminler) elde edilir. MFFs'nin girdi matrisi, eğitilmiş yöntemlerin sonuçlarının toplamıdır.

$$Z = [Z_{ij}] , i = 1,2 \dots n_{test}, ; j = 1,2, \dots, m$$

Burada Z_{ij} , j'inci yöntem için i'inci veri noktasının sonuçlarıdır.

$$Z = \begin{bmatrix} Z_{1,1} & \dots & Z_{1,n_{test}} \\ \vdots & \ddots & \vdots \\ Z_{m,1} & \dots & Z_{m,n_{test}} \end{bmatrix}$$

Adım 5: Girdi matrisi Z ikiye bölünmüştür, eğitim Z_{train} ve test Z_{test} kümeleri. Eğitim seti, fonksiyonlardaki yöntemlerin ağırlıklarını belirlemek için, test seti ise MFFs'lerin performansını değerlendirmek için kullanılır.

$$Z_{train} = [Z_{ij}] ; i = 1,2 \dots, n_{train1} ; j = 1,2, \dots, m$$

$$Z_{ntest} = [Z_{ij}] ; i = n_{train1}, n_{train1+1}, \dots, n_{test} ; j = 1,2, \dots, m$$

Adım 6: Girdi matrisi Z , FCM kullanılarak kümelendirilmiştir. Her kümedeki üyelik dereceleri, fonksiyonlardaki yöntemlerin ağırlıklarını belirlemek için kullanılır. Bu durumda, bir küme bir fonksiyonu temsil eder.

Adım 6.1: Bulanıklık indeksi parametresini, küme sayısını ve ilk küme merkezlerini belirleyin.

Literatürde bulanıklık parametresinin (m) seçimine dayalı olarak araştırmacılar tarafından birçok çalışma yapılır. Bunlardan biri, m 'nin optimal değerinin [1.5,2.5] ile sınırlı olduğuna işaret eden Pal ve Bezdek (1995) tarafından tanıtılır. Özkan ve Türksen (2007) ise m 'nin üst ve alt değerinin sırasıyla 1,4 ve 2,6 olduğunu tespit eder. Chan ve Cheung (1992), kelime tanıma çalışmasında m 'nin değerinin olması gerektiğini öne sürer. Ayrıca Bezdek (1981), optimum seçimin 2 olduğunu öne sürer.

Adım 6.2: Denklem (3.40) ile üyelik değerini hesaplayın

$$\mu_i = \left[\frac{\sum_{k=1}^n d(z_k, v_i)^{\frac{2}{m-1}}}{\sum_{k=1}^n d(z_k, v_j)^{\frac{2}{m-1}}} \right]^{-1} ; i = 1, 2, \dots, c ; k = 1, 2, \dots, n \quad (3.40)$$

$\sum_{i=1}^c \mu_{ik} = 1$ kısıtı altında eğer $\mu_{ik} < \alpha$ -kesim ise; daha sonra değer sıfır olarak alınacaktır. Z Girdi matrisi, v küme merkezleridir, $d(\cdot)$ Öklid uzaklık fonksiyonu anlamına gelir, c küme sayısıdır ve m (3.40)- (3.41) denklemlerindeki bulanıklık parametresidir.

Adım 6.3: Yeni küme merkezlerini hesaplayın.

$$v_i = \frac{\sum_{k=1}^n \mu_{ik}^m z_k}{\sum_{k=1}^n \mu_{ik}^m} \quad (3.41)$$

Adım 6.4: İki yineleme arasındaki küme farkı bir eşğin altına düşene veya yineleme sayısına ulaşılan kadar Adım 6.2 ve Adım 6.3'ü tekrarlayın.

Adım 7: Denklem (3.42)'de verilen MFF'ler eğitim seti için hesaplanan üyelik dereceleri kullanılarak elde edilir.

$$MFF_i(z) = \sum_{j=1}^m w_{ij} z_j ; i = 1, 2, \dots, c \quad (3.42)$$

$$w_{ij} = \frac{\mu_{ij}}{\sum_{j=1}^m \mu_{ij}} ; i = 1, 2, \dots, c \quad (3.43)$$

burada MFF_i i 'nci meta bulanık fonksiyonunu temsil ederken μ_{ij} , i 'nci kümedeki j 'inci yöntemin üyelik değerinin derecesini ve c küme sayısını temsil eder.

Adım 8: Farklı m ve c 'ler için Adım 6–7'yi tekrarlayın.

Adım 9: Bir fonksiyondaki yöntemlerin en iyi kombinasyonunu seçme. En iyi değerlendirme kriterlerine sahip fonksiyon, en iyi meta bulanık fonksiyonu (MFF_{best}) olarak seçilir.



Adım 10: MFF sisteminin sonuçları, test seti Z_{test} için MFF_{best} kullanılarak hesaplanır.

$$F = MFF_{best}(Z_{test}) \quad (3.44)$$

Burada F , MFF'lerin sonuçlarıdır.



4. KARMA ARAÇ ROTALAMA ALGORİTMALARI

Karma Araç Rotalama Problemi (KARP), temelde Araç Rotalama Problemi (ARP) olup Dantzig ve Ramser (1959) tarafından kamyon sevkiyatı problemi hakkındaki ilk makalenin yayınlanması ile ortaya atılan ve günümüzde dağıtım ve ulaşım alanlarında yaygın olarak kullanılan bir kombinatoriyal optimizasyon problemidir.

KARP, kombinatoriyal optimizasyonda NP-Zor bir problem olarak kabul edilir. Bu, sorunun boyutu arttıkça, hesaplama karmaşıklığının katlanarak arttığı anlamına gelir. Orta ölçekli KARP problem verilerinde bile, büyük sayıda değişken kullanılabilir. KARP'nin NP-Zor olması nedeniyle düğüm sayısının fazla olduğu durumlarda makul hesaplama süresi içinde optimum turları verecek kesin bir algoritma garanti edilemez. (Cordeau vd. 2006). Bu nedenle, bu tür problemleri çözmek için kesin yöntemlere göre çözüm süresi daha kısa olan sezgisel algoritmalar kullanılabilir.

4.1. Geleneksel karma araç rotalama modeli

Geleneksel KARP'da talep noktalarının her biri aynı önem ağırlıklarına sahiptir ve model iki temel kararı içerir. Bunlar rotalanacak aracın seçimi ve bu aracın rotasıdır. Araç filosu farklı kapasite ve maliyetlere sahip araçlardan oluşabilir. KARP, bir veya daha fazla depodan yönlendirilen karma bir araç filosu için rotalar tasarlayarak, coğrafi olarak dağınık bir dizi müşteriye nasıl hizmet verileceğini belirler. Güzergahlar tamamlandığında araçlar aynı depoya geri döner. Dolayısıyla KARP'ların amacı, kat edilen toplam mesafeyi en aza indirerek rotaları optimize etmek ve böylece tüm kısıtlamalara uyarak maliyetleri azaltmaktır. Bu amaca, en uygun araç karışımını bularak ve problem kısıtlamalarını karşılarken ilgili rotaları belirleyerek ulaşılabilir. (Renaud ve Boctor 2002).

KARP matematiksel modelinde kullanılan değişkenler ve parametreler aşağıda verilmiştir:

Parametreler

N	Talep noktaları kümesi
K	Araç seti
B	Başlangıç noktası

d_{ij} i ve j noktaları arasındaki Öklid uzaklığı
 f_k k tipi aracın menzili

Karar Değişkenleri:

x_{ijk} i noktasından j noktasına k aracı ile gidiliyorsa 1, aksi takdirde 0 değerini alan ikili değişken

y_i Alt turları önlemek için yardımcı değişken

KARP bir kombinatoriyal optimizasyon ve tamsayı programlama problemidir ve temel KARP modelini aşağıdaki gibi sunuyoruz:

$$\text{Min } \sum_{i \in N \cup \{B\}} \sum_{j \in N \cup \{B\}} \sum_{k \in K} d_{ij} x_{ijk} \quad (4.1)$$

$$\sum_{i \in \{B\} \cup N \setminus \{j\}} x_{ijk} = \sum_{i \in \{B\} \cup N \setminus \{j\}} x_{jik} \quad j \in N \cup \{B\}, k \in K \quad (4.2)$$

$$\sum_{i \in N \cup \{B\}} \sum_{k \in K} x_{ijk} = 1 \quad j \in N \quad (4.3)$$

$$\sum_{i \in N \cup \{B\}} \sum_{j \in N \cup \{B\}} x_{ijk} d_{ij} \leq f_k \quad k \in K \quad (4.4)$$

$$y_i - y_j + (|N \cup \{B\}| - 1)x_{ijk} \leq |N \cup \{B\}| - 2 \quad i, j \in N \quad i \neq j, k \in K \quad (4.5)$$

$$1 \leq y_i \leq |N \cup \{B\}| - 1 \quad i \in N \quad (4.6)$$

$$x_{ijk} \in \{0,1\} \quad i, j \in N \cup \{B\}, k \in K \quad (4.7)$$

Amaç fonksiyonu (4.1), ziyaret edilen talep noktaları arasındaki Öklid mesafelerinin toplamını en aza indirir. Kısıt (4.2) aynı araçla bir noktaya giriş ve çıkış sağlar. Kısıt (4.3), her talep noktasının bir kez ziyaret edilmesini garanti eder. Kısıt (4.4) her aracın kat ettiği toplam mesafenin aracın menzilinden daha az olmasını sağlar. Kısıtlar (4.5)-

(4.6), alt turları engelleyen kısıtlamalardır. Kısıt (4.7), karar deęiřkenlerinin 0-1 tamsayı olmasını saęlar.



Karmaşıklığı nedeniyle, KARP ile ilgilenen araştırmalar sezgisel yöntemleri de ele alır. Bu alana en önemli katkılardan biri Golden vd. (1984) tarafından Clarke ve Wright'ın (1964) tasarruf algoritmasının yeni beş uyarlamasını getirerek sağlanır. Gould (1969), KARP için sadece depo ile her müşteri arasındaki gidiş dönüşlerin dikkate alındığı doğrusal bir program geliştirir. Woods ve Harris (1979), bir simülasyon yaklaşımı kullanarak KARP modelini ele alır. Etezadi ve Beasley (1983), araçların birçok müşteriyi ziyaret edebileceği bir formülasyon sunar. Golden vd. (1984), KARP için matematiksel bir formülasyon sunar. Çalışmada, problemi çözmek için farklı bir strateji olarak, elde edilen çözümü tekrar tekrar iyileştiren Tabu Arama yöntemi önerilir. Gendreau vd. (1999), Gezgin Satıcı Problemini çözmek için kendileri tarafından geliştirilen GENIUS (Gendreau vd. 1992) adlı sezgisel yöntemi uyarlayarak uyarlanabilir bellek prosedürüne sahip Tabu Arama tabanlı yeni bir sezgisel yöntem önerir. Bir dizi çözüm üretmek için Tabu Araması'nı kullanan farklı bir yaklaşım sunar ve en iyi rota kombinasyonunu seçmek için bir Set Bölümleme Problemini çözer. Bu sezgisel yöntem mükemmel sonuçlar vermesine rağmen, kullanılan Set Bölümleme Probleminin NP zor bir problem olduğunu ve bu nedenle çözülmesi çok zor olabileceğini ve büyük problemler için oldukça zaman alabileceği fark edilir.

Taillard (1999) tarafından, karma bir araç filosu ile araç yönlendirme sorunlarını çözmek için sezgisel bir Sütun Türetme Yöntemi sunulur. Yöntem aynı zamanda filo büyüklüğü ve bileşimi yönlendirme problemini çözebilmekle birlikte bir dizi klasik problem için de en iyi bilinen çözümler rapor edilir. Sayısal sonuçlar, yöntemin özellikle orta ve büyük boyutlu problem örnekleri için sağlam ve verimli olduğunu göstermektedir.

Renaud ve Boctor (2002), çalışmasında karma araç filosu rotalama problemi için yeni bir süpürme tabanlı sezgisel tarama sunmaktadır. Önerilen algoritma önce bir veya iki araç tarafından servis edilen çok sayıda rota üretir. Kullanılacak rotaların ve araçların seçimi, polinom zamanda, özel bir yapıya sahip bir küme bölme probleminin en iyi duruma getirilmesiyle çözülür. Bir dizi kıyaslama testi probleminin sonuçları, önerilen yöntemin kısa hesaplama sürelerinde mükemmel çözümler ürettiğini göstermektedir. Filolarının önemli bir bölümünü kiralayan ve sonuç olarak filo bileşimindeki sık değişikliklerden faydalanabilecek nakliye şirketleri için hızlı ama iyi bir çözüm

yöntemine ihtiyaç vardır. Çalışmada, önerilen sezgisel tarama üç test verisi için en iyi bilinen yeni çözümler üretir, bu çözümler rapor edilir.

Dondo ve Cerda (2007) çalışmasında, zaman pencereleri ve karma araç filosu ile çoklu depo yönlendirme problemi için yeni bir üç fazlı sezgisel / algoritmik yaklaşım sunmaktadır. Sezgisel tabanlı bir kümeleme algoritmasının bir Zaman Pencereli Araç Rotalama Problemi (ZP-ARP) optimizasyon çerçevesi içine yerleştirilmesinden türetilir. Bu amaçla, ilk olarak ZP-ARP için titiz bir Karma Tamsayılı Doğrusal Programlama matematiksel modeli tanıtılır. Diğer optimizasyon yaklaşımları gibi, yeni formülasyon da en fazla 25 düğümü içeren vaka çalışmalarını verimli bir şekilde çözebilir. Bu sınırlamanın üstesinden gelmek için, başlangıçta daha kompakt bir kümeyle dayalı Karma Tamsayılı Doğrusal Programlama formülasyonu elde etmek için bir ön işleme aşaması kümeleme düğümleri birlikte gerçekleştirilir. Bu şekilde, bir sezgisel ve iki algoritmik faz içeren bir hiyerarşik hibrit prosedür geliştirilir. Faz I, araçlara kümeler atar ve küme tabanlı Karma Tamsayılı Doğrusal Programlama formülasyonunu kullanarak her turda bunları dizilerken bir dizi uygun maliyetli uygulanabilir küme belirlemeyi amaçlamaktadır. Küçük bir Karma Tamsayılı Doğrusal Programlama modelini çözerek kümeler içindeki düğümleri sipariş etmek ve her bir tur için müşteri yerlerine araç varış zamanlarını planlamak nihayet Faz II'de gerçekleştirilir. Farklı boyutlar, kümelmiş / rastgele müşteri konumları ve zaman penceresi dağılımları içeren çok sayıda kıyaslama problemi kabul edilebilir zamanlarda çözülür.

Azi vd. (2010) çalışmalarında birden fazla aracın bulunduğu problemleri ele alır ve bu problem için bir kesin algoritma olan Dal-Fiyat Algoritması önerir. Problem boyutlarındaki kısıtlamalara ve problemin karakteristik sınırlamalarına rağmen önerilen Dal-Fiyat algoritması, 25 müşterilik problemi rahatlıkla çözebilmiş ve birkaç istisnaya 50 müşterilik problemlere kadar en iyi çözümlere ulaşılır.

Belmecheri vd. (2013), çalışmalarında Zaman Pencereli Karma Araç Filosu Rotalama Problemini (ZP-KARP) ele alır. Çalışmada, yerel arama ile bir Parçacık Sürüsü Optimizasyonu (PSO) önerilir. Bu yaklaşımın çeşitli kombinatoriyal problemler üzerindeki etkinliğini gösterilir ve incelenen probleme uyarlanması karşılaştırmalı olarak açıklanmış ve test edilir. Sonuçlar önceki yöntemlerle karşılaştırılır ve birkaç durumda PSO'nun sonuçları iyileştirdiği gösterilir.

Jiang vd. (2014), sınırlı sayıda karma bir filo ile ZP-KARP ele alır. Çalışmada problemi çözmek için mevcut bir tabu arama prosedürünü genişleten bir yöntem önerilmektedir. Önerilen yöntemin performansını değerlendirmek için, karma bir filo ile araç yönlendirme probleminin çok sayıda varyantından çeşitli karşılaştırma problemlerini içeren geniş bir test senaryosu seti üzerinde deneyler yapılır. Önerilen yöntemin, bu problem varyantları için makul derecede iyi sonuçlar vermek için kullanılabilmesi gözlenir. Buna ek olarak, algoritmik karşılaştırma için geliştirilen sezgisel araştırmaları iletirmek için bazı fikirler sunulmaktadır.

Sassi vd. (2015), çalışmalarında geleneksel ve elektrikli araçlardan oluşan karma araç filosu ile araç rotalama problemini uygulamalı olarak inceler. Çalışmada ele alınan problem, farklı akü kapasitelerine ve işletim maliyetlerine sahip karma elektrikli araç filosu ve bir dizi aynı tip konvansiyonel araç içeren filo tarafından hizmet sunulması gereken bir dizi müşteri içermektedir. Çalışmadaki elektrikli araçlar belirli bir şarj teknolojisi ve zamana bağlı şarj maliyetleri ile şarj sunan mevcut şarj istasyonlarındaki seyahatleri sırasında şarj edilebilir durumda olup şarj istasyonları çalışma süresi ayrıca pencere kısıtlamalarına tabidir. Amaç, kullanılan araç sayısını en aza indirmek ve toplam seyahat ve şarj maliyetlerini en aza indirmektir. Geliştirilen çoklu başlatma algoritması, Yerel Arama prosedüründe iki farklı ekleme stratejisiyle Büyük Komşuluk Arama kullanan Yinelemeli Yerel Arama meta-sezgisel yöntemine dayanır. Önerilen yöntemin farklı uygulamaları, 550 müşteriye kadar bir dizi gerçek veri örneğinde test edilir.

Cattaruzza vd. (2016), çok rotalı araç rotalama problemi ve araçların birden çok yolculuk yapmasına izin verilen rotalama problemi hakkında bir araştırma sunmaktadır. Çalışmanın ilk kısmı çok rotalı araç rotalama problemine odaklanmaktadır. Bu aşamada problem çözümü için matematiksel formülasyonlar üzerinde entegre bir bakış sunulur ve kesin ve sezgisel yaklaşımlar ile çözüm incelenir. Çalışma, bazen birden fazla seyahate izin verilen diğer yönlendirme problemleri aileleriyle devam etmektedir. İkinci problem için, özellikle birden fazla yolculuk yapma motivasyonları ve algoritmik sonuçları konusunda araştırma yapılır. Çalışmanın beklenen katkısı, rotalama probleminin yapısal özellikleri hakkında kapsamlı bir genel bakış sağlamaktır.

Koç vd. (2016) çalışmalarında karma araç filosu rotalama problemi ve varyantlarının ortaya çıkmasından bu yana yaklaşık 30 yıllık süreçte ele alınan çalışmaları değerlendirir. Bu araştırma belgesinin amacı karma araç filosu rotalama problemleri hakkındaki literatürü sınıflandırmak ve gözden geçirmektir. Bu makale aynı zamanda bu problemler için önerilen meta-sezgisel algoritmaların karşılaştırmalı bir analizini sunmaktadır.

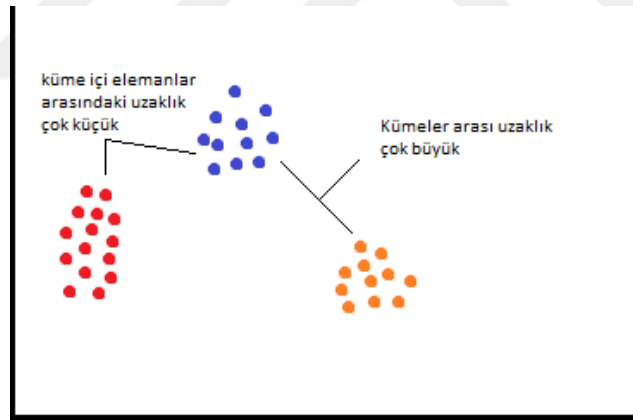
Wang vd. (2016), dronlarla araç rotalama problemini ele alır. Çalışmada, dronlarla donatılmış bir kamyon filosu müşterilere paketler sunar. Dronlar depodaki kamyonlardan veya herhangi bir müşteri lokasyonundan gönderilebilir ve alınabilir. Amaç, rotaların maksimum süresini en aza indirmektir (yani tamamlanma süresi). Dronlu Araç rotalama problemi, Amazon, DHL ve Federal Express gibi bir dizi oldukça etkili şirket tarafından teşvik edilmektedir ve paket teslimatı için ticari dronların potansiyel kullanımını oldukça yaygındır. Çalışmada belirli varsayımlar altında, dronları kullanarak elde edilebilecek maksimum kazancı incelemek için bir dizi soru sorulur. Daha sonra bir dizi en kötü durum sonucu elde edilir. En kötü durum sonuçları, kamyon başına dron sayısına ve dronların kamyon hızına göre hızına bağlıdır.

Macrina vd. (2019), Elektrikli ve konvansiyonel (içten yanmalı motor) araçlardan oluşan karma araç filusunun rotalanmasını ele alan Yeşil Araç Yönlendirme Probleminin belirli bir versiyonunu araştırır. Çalışmada ele alınan araçlar tipik olarak hafif ve orta hizmet araçlardır. Çalışmada, mevcut istasyonların herhangi birinde kısmi pil şarjına izin verilen ve hızlanma, yavaşlama vb. kriterleri dikkate alabilecek büyük bir komşuluk arama planına gömülü bir meta-sezgisel önerilir. Aynı zamanda çalışmada sayısal örneklerle çalışmada önerilen yaklaşımın performansı değerlendirilir.

5. KÜMELEME ALGORİTMALARI

Kümeleme kavramı, en basit tanımıyla birbiriyle benzer özellik taşıyan veri elemanlarının kendi içlerinde gruplara ayrılmasıdır. Bir dizi veri noktası göz önüne alındığında, nesneleri temel olarak sahip oldukları özelliklere göre gruplandırmak için kullanılan çok değişkenli bir tekniktir. Günümüzde kümeleme algoritmaları birçok alanda uygulanmaktadır (Wu vd. 2014).

Kümeleme algoritmalarının amacı, küme içindeki homojenliği sağlarken kümeler arasını heterojen yapmaktır. Kümeleme algoritmalarında, aynı kümede yer alan veriler arasındaki uzaklık çok azken kümeler arası uzaklık fazladır. Bu da benzer verilerin aynı küme içerisinde olmasıyla sağlanabilir. Buradaki benzerlik uzaydaki konum noktaları ile ilgilidir. Konum noktaları itibariyle birbirleri arasındaki mesafe daha az olan veriler aynı kümede toplanır. Şekil 5.1.'de gösterilir.

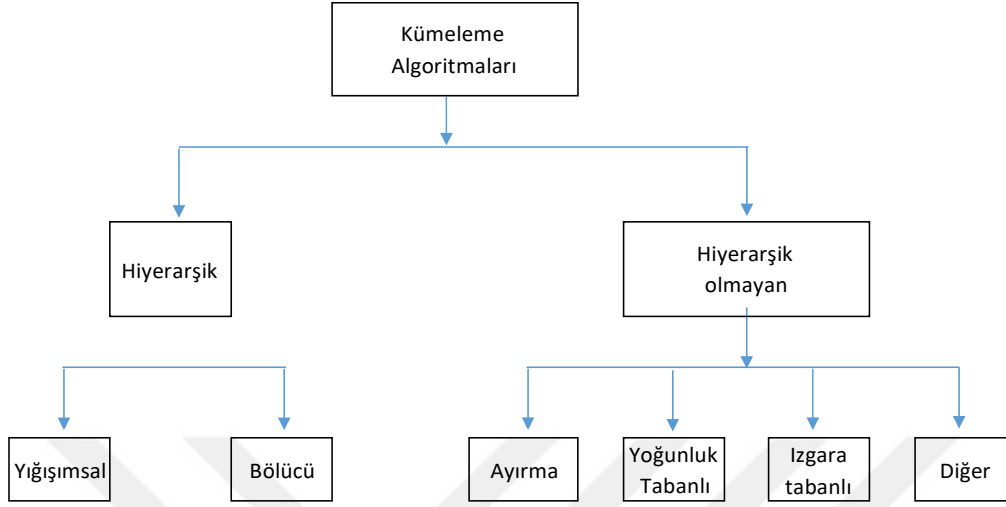


Şekil 5.1 : Kümeleme gösterimi.

Veriler arasındaki uzaklıkların hesaplanmasında kullanılan yaygın yöntemler Minkowski, Öklid (Euclidean), Pearson, Manhattan (City-Blok), Mahalanobis, Hotelling T2 ve Canberra Uzaklığı'dır (Arslan, 2008).

Mevcut kümeleme algoritmaları genel olarak hiyerarşik ve hiyerarşik olmayan kümeleme algoritmaları olarak sınıflandırılabilir (Jain ve Dubes, 1988). Kümeleme algoritmalarından hangisinin kullanılacağı verinin türüne, kullanım amacına ve uygulamaya bağlıdır. Bazı problemlerde veri seti üzerinde birden fazla algoritmanın

uygulanması gerekebilmektedir (Akın, 2008). Kümeleme yaklaşımları literatürde Şekil 5.2'deki gibi sınıflandırılmaktadır (Eden ve Tommy 2004).



Şekil 5.2 : Kümeleme algoritmaları.

Hiyerarşik olmayan sınıfa giren kümeleme algoritmaları veriyi doğrudan kümeleyen algoritmalarıdır. Ayırma tipi kümeleme, hiyerarşik olmayan kümeleme sınıfında en yaygın kullanılan yaklaşımdır. Bu tip algoritmalar genellikle, tüm noktalar ilgili küme merkezleri ile uzaklıkları minimuma ininceye kadar kümelerin merkezini değiştirir. Ayırma yaklaşımın en yaygın örneği olarak “K-Means” verilir. “Bulanık C-Ortalama”, “K-modes” ve “K-prototip” gibi algoritmalar “K-Means” algoritmasının çeşitleridir.

Hiyerarşik sınıf ile karşılaştırıldığında ayırma yaklaşımı hesaplama zamanı bakımından daha düşük maliyetlidir. Ancak, tüm noktaların merkezleri ile mesafesi minimum olana kadar, her nokta ile ilgili merkezi arasındaki mesafe ardışık olarak hesaplandığı için yine de yüksek maliyetli bir süreçtir.

5.1 K-Means kümeleme algoritması

K-Means kümeleme, bir veri kümesini otomatik olarak K gruba bölmek için yaygın olarak kullanılan bir yöntemdir (MacQueen, 1967). Khanmohammadi vd. (2017)'ye göre K-Means algoritması bir centroid modeldir. Centroid modu, kümeler oluşturmak için centroid kullanan bir modeldir ve Centroid, bir kümenin orta noktasıdır yani bir değerdir. Centroid, bir veri nesnesinin centroid ile olan mesafesini hesaplamak için kullanılır (Na vd. 2010).

Algoritma temelde aşağıdaki adımlarla ilerler :

Adım 1: Yineleme sayacı t , 1 olsun. Yineleme t 'deki küme merkezleri $m_1^{(t)}, m_2^{(t)}, \dots, m_k^{(t)}$. olsun. $t=1$ için ilk küme merkezleri rastgele belirlenir.

Adım 2: Her gözlem, Öklid mesafesine göre en yakın kümeye (tek bir kümeye) atanır ve kümeler Denklem (5.1)'deki gibi oluşturulur.

$$S_i^{(t)} = \{x_j : \|x_j - m_i^{(t)}\|_2 \leq \|x_j - m_j^{(t)}\|_2 \forall j = 1, 2, \dots, k\} \quad (5.1)$$

Adım 3: 2. adımda yapılan atamalara göre yeni merkezler Denklem (5.2)'deki gibi belirlenir.

$$m_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{x_j \in S_i^{(t)}} x_j \quad (5.2)$$

Adım 4: Küme merkezleri stabilize olduysa, durun. Aksi takdirde, t 'yi 1 artırın ve Adım 2'ye dönün.

Matematiksel ifade olarak, her bir veri n boyutlu reel vektör olmak üzere bir $\{x_1, x_2, \dots, x_N\}$ veri kümesi ve bölünecek küme sayısı olarak K verilsin. K-Means kümeleme, Hata Karelerinin Toplamı'nı en aza indirmek için N tane veriyi K adet $S = \{S_1, S_2, \dots, S_K\}$ kümeye bölmeyi amaçlar.

Amaç fonksiyonu matematiksel olarak (5.3) ve (5.4)'deki şekilde ifade edilebilir:

$$\operatorname{argmin} \sum_{j=1}^K \sum_{x_i \in S_j} \|x_i - \mu_j\|_2^2 \quad (5.3)$$

$$\mu_j = \frac{1}{|S_j|} \sum_{x_i \in S_j} x_i \quad (5.4)$$

Burada; μ_j, S_j 'deki noktaların ortalamasıdır.

6. GÖÇMEN SAYISI TAHMİNİ VE DEVRİYE BOTU ROTALAMASI

Deniz hattında gerçekleşen göç olayları insan, uyuşturucu veya tarihi eser kaçakçılığı amaçlı olabilir. Her durumda, bir güvenlik tehdidi oluşturur ve tespit edilmesi kritik öneme sahiptir.

Deniz hattı güvenliğini sağlamak için kıta sahanlığı içinde sürekli devriye gezilmesi gerekir. Ancak deniz hattında güzergah alternatifleri sınırsız iken kaynaklar sınırlıdır ve ziyaret edilmesi gereken her noktada sabit bir devriye olması mümkün ve pratik değildir. Bu nedenle devriye aracı rotalaması zor bir problemdir ve verimli ve etkili yaklaşımlar gerektirir.

Göç hareketinin öngörülebilir olması, Sahil Güvenlik Komutanlığı'nın verimli bir devriye rotası izlemesine yardımcı olacak ve göçmenlerin tespitini artırır. Böyle bir yaklaşımı sağlamak için, deniz hattı güvenliği projesinde, bundan böyle sıcak noktalar olarak anılacak olan ve farklı kritiklik seviyelerine sahip olabilecek talep noktalarını K-Means kümeleme algoritması ile elde ediyoruz. Ardında sıcak noktalara ait göçmen sayılarını Meta Bulanık Fonksiyon Tabanlı Tahmin Kombinasyonu Yaklaşımı (FCA-MFFs) ile elde ediyoruz ve bu tahmin sonuçlarını sıcak noktaların önem ağırlığı olarak alıyoruz. Son olarak sınırlı devriye araçlarının varlığında önem ağırlıkları farklı olan bu sıcak noktaların ziyaret edilebilecekleri rotalama çözümleri öneriyoruz.

6.1 Problemin Tanımlanması

Türkiye, jeopolitik konumu itibarıyla karayolu ve deniz yolu üzerinden yoğun olarak göç almakta veya göç vermektedir. Göç olayı demografik etkilerinin yanı sıra ekonomik, siyasi ve kültürel açıdan pek çok olumsuz etkiye neden olmaktadır. Bu etkilerden en önemlisi; Sahil Güvenlik Komutanlığı'nın deniz yolu üzerinde sürekli devriye yapmasına neden olarak ülke kaynaklarının verimsiz kullanılmasına ve maddi zararlara sebep olmasıdır.

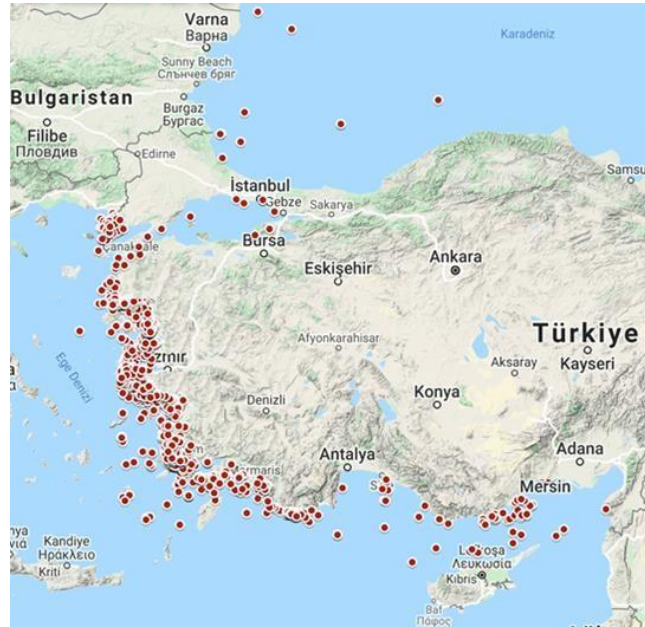
Türkiye'de deniz yolu üzerinde gerçekleşen göç olayları çoğunlukla Türkiye'den çıkış yapacak olan göçmenlere ait olsa da, bu göçlerin tespit edilmesi yine de kritiktir. Öncelikli olarak mevcut göçmenlerin can kaybı yaşamaması istenmez, dolayısıyla sağ salim tespit edilerek ülkelerine iade edilmeleri veya siyasi olarak gereğinin yapılması

istenir. Ek olarak kaçakçılık amacı ile yapılan göçlerin de tespit edilmesi ve gereğinin yapılması önemlidir. Dolayısıyla her durumda göç hareketinin tespit edilmesi büyük önem arz etmektedir.

Göç hareketinin önceden tespit edilmesiyle Sahil Güvenlik Komutanlığı'nın verimli bir devriye rotası izlemesi sağlanarak yakalanan göçmen sayısının artırılması hedeflenir. Bu uygulamanın devriye maliyetlerini azaltmasının yanı sıra caydırıcı olma niteliği ile göç hareketlerini de azaltması hedeflenmektedir.

6.1.1 Problemin verileri

Sahil Güvenlik Komutanlığı tarafından 1 Ocak 2015 ile 31 Ağustos 2019 tarihleri arasında deniz yolu üzerinde tespit edilen göçmen verilerine ait Derece-Dakika-Saniye (DMS) cinsinden Enlem ve Boylam bilgileri gizlilik politikası nedeniyle paylaşamamaktadır. Ancak bu veriler analiz edildiğinde göçmen tespit edilen enlemlerin 26°, 32°, 35°, 36°, 37°, 38°, 39°, 40°, 41°, 42° ve 43° üzerinde olduğu görülürken boylamların 25°, 26°, 27°, 28°, 29, 30°, 31°, 32°, 33°, 34° ve 35° olduğu tespit edilmiştir. Bu verilerden yola çıkarak göçmen tespit edilen noktalar Şekil 6.1.'de Türkiye'nin coğrafi haritası üzerinde temsili olarak gösterilmektedir.



Şekil 6.1 : Göçmen yakalanan koordinatların harita üzerinde gösterimi.

Şekil 6.1.'de görüldüğü üzere, Türkiye'nin sınırı olan tüm denizlerde; Ege Denizi, Akdeniz, Karadeniz ve Marmara Denizi; göçmen tespiti yapılmış olup bu denizler üzerindeki rotaların potansiyel göç rotası olarak alınmasını önemli görüyoruz.

Ancak gerek denizlerin akıntı yapısı, gerekse buldukları bölgeye bağlı olarak değişen coğrafik şartları nedeniyle göçmen barındırma potansiyelleri farklıdır. Bu nedenle göçmen sayısı tahmininin tüm veriler üzerinden yapılmaması, verilerin gruplara ayrılarak belirlenen her bir grubun kendi içinde ayrı ayrı tahmin yapılması gerektiğine karar verilir.

Gruplama yapılırken öncelikle farklı alternatifler değerlendirilir. Bu alternatifler ilk aşamada aşağıdaki şekilde belirlenir:

- *Alternatif 1* : Muhtemel kalkış yeri ve muhtemel varış yeri aynı olan verilerin aynı grup altında toplanması,
- *Alternatif 2* : Aynı enlem derecesine sahip verilerin aynı grup altında toplanması,
- *Alternatif 3* : Aynı boylam derecesine sahip verilerin aynı grup altında toplanması,
- *Alternatif 4* : Aynı enlem ve boylam derecesine sahip verilerin aynı grup altında toplanması,
- *Alternatif 5* : Aynı denizde yer alan verilerin aynı grup altında toplanması,
- *Alternatif 6* : Göç yoğunluğuna göre verilerin aynı grup altında toplanması.

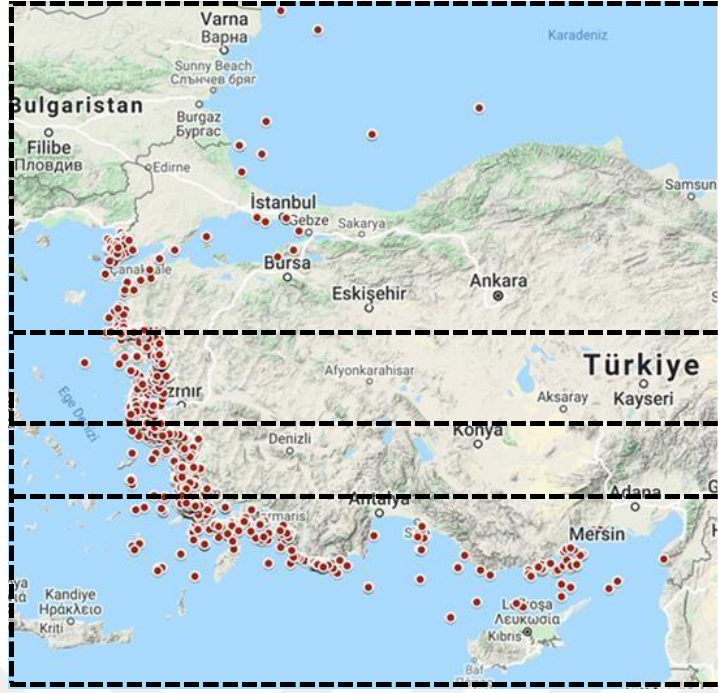
En iyi alternatifi seçmek amacıyla veriler incelendiğinde en çok göçmen kaçıışının Ege Denizi üzerinde gerçekleştiği tespit edilir. Bu noktada öncelikle göç olayının mantığını ve gerçekleştirilme biçimini anlamakta fayda vardır. Bölüm 1'de detaylı olarak açıklandığı üzere deniz yolu üzerinde gerçekleşen göçlerin Türkiye'de iki temel sebebi vardır. İlki kaçakçılık amacıyla yapılan göçlerdir. İkincisi, Doğu ülkelerdeki savařlardan kaçıp Türkiye'ye sığınan göçmenlerin Batı ülkelere geçiş yapmak amacıyla başvurduğu göç tipidir. Ege Denizi'ne kıyısı olan komşu ülke Yunanistan ve ona bağlı adalar göz önüne alındığında her iki durumda da göçlerin Türkiye'nin batı kıyılarından Yunanistan'a doğru olması yüksek ihtimaldir. Dolayısıyla Türkiye'nin jeopolitik konumu gereği göçmenlerin kaçış rotaları enlemsel bir doğrultuda olacaktır. Buna paralel olarak devriye botlarının rotasının da enlemsel olması mantıklıdır ve bu

rota üzerindeki devriye Türkiye'nin kıyı illerinden Yunanistan'a doğru bir göçmen kaçıışı var ise yakalamayı kolaylaştırır. Aynı zamanda devriye botunun dönüşünde Yunanistan'dan Türkiye'ye gerçekleşen bir göç olayı varsa yine tespit edilmesi mümkün olacaktır. Dolayısıyla göçmen tespitini verimli yapabilmek için devriye botu güzergahlarının enlemsel bir hareket yaparak dolaşması gerektirmektedir.

Alternatif 3 aynı boylama sahip verilerin gruplanmasını önermektedir, ancak detaylı olarak açıkladığımız üzere boylamsal bir göç rotası mantıklı olmadığı için boylamsal bir inceleme yapılmasına gerek yoktur ve bu tür gruplamanın verimsiz olması sebebiyle Alternatif 3 elenir.

Alternatif 5 değerlendirilecek olursa; mevcut veriler analiz edildiğinde Ege Denizi üzerinde göç olayının yoğun olmasının yanı sıra Akdeniz, Karadeniz ve Marmara denizi üzerinde gerçekleşen göç olayları daha seyreklerdir. Dolayısıyla bu denizler üzerinde gerçekleşen göç verileri azdır. Az veri üzerinden gelecek tahmini yapmak anlamlı olmamaktadır. Bu da Alternatif 5'in elenmesini sağlar.

Geriye kalan Alternatif 1, Alternatif 2, Alternatif 4 ve Alternatif 6 verimli gruplamalardır. Hepsi de enlemsel harekette yapılacak devriye rotalamalarında tespitte yardımcı kümeleri vermektedir. Muhtemel kalkış yeri ve muhtemel varış yeri aynı olan verilerin aynı grup altında toplanması, aynı enlem derecesine sahip verilerin aynı grup altında toplanması veya aynı enlem ve boylam derecesine sahip verilerin aynı grup altında toplanması ile enlemsel bir düzlemde veri setleri oluşturulmuş olur. Göç yoğunluğuna göre verilerin aynı grup altında toplanması ile de tahmin metodu geliştirmek için yeterli sayıda veri seti içeren gruplar elde edilmiş olunur. Dolayısıyla bu dört alternatifi birleştiren yeni bir alternatif önerilmiş olup Şekil 6.2'de belirtilen şekilde gruplama yapmak anlamlı bulunur.



Şekil 6.2 : Bölgelerin gruplandırılması gösterimi.

Göçmen tespiti yapılan koordinatların enlemleri 26° , 32° , 35° , 36° , 37° , 38° , 39° , 40° , 41° , 42° ve 43° olarak tespit edilir. Mevcut enlemlerin göç yoğunlukları göz önüne alınarak ve alanında uzman Sahil Güvenlik Komutanlığı personellerinin görüşlerine dayanarak tüm verilerin Çizelge 6.1'deki enlemler altında gruplanmasına karar verilir ve geleceğe yönelik göçmen sayısı tahmini yapılacak dört adet veri seti elde edilir.

Çizelge 6.1 : Enlem-bölge eşleştirmesi.

ENLEM	BÖLGE	
35°	Bölge-1	Kıta Sahaneliği Bölgesi
36°		
37°	Bölge-2	
38°	Bölge-3	
39°	Bölge-4	
40°		

Mevcut göç verileri günlük bazlıdır. Aynı zamanda bazı günler farklı saatlerde olmak üzere birkaç kez göçmen tespit edilir. Haftalık göç tahmini yapabilmek için veriler haftalık göçmen sayısını verecek şekilde dönüştürülür. Bölge bazında haftalık olarak tespit edilen toplam göçmen sayısı bilgileri ÇizelgeEk-1'de verilir.

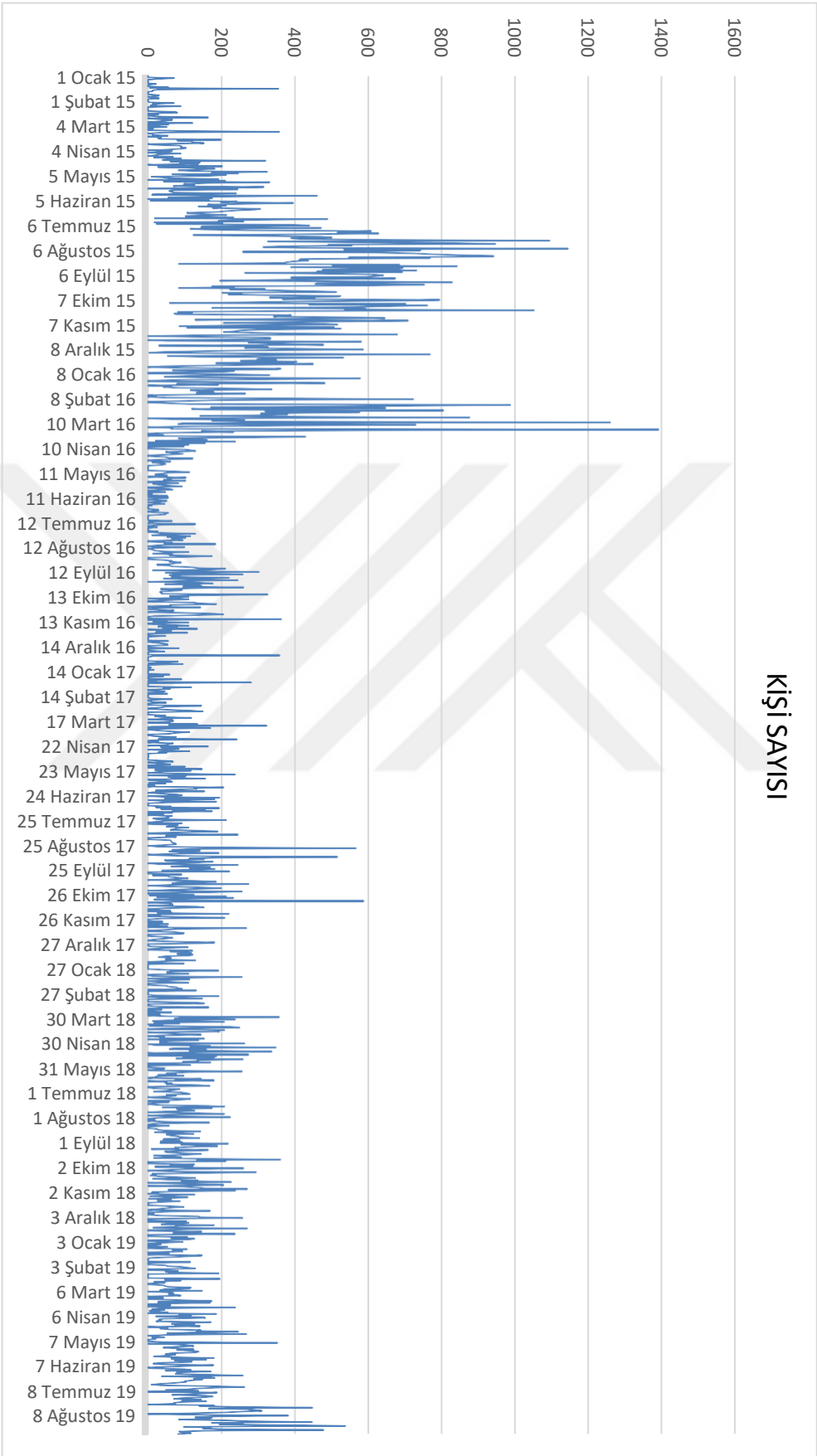
Sahil Güvenlik Komutanlığı envanterinde bulunan ve devriye botu rotalaması için kullanılan araçlara ait bilgiler Çizelge 6.2’de verilir.

Çizelge 6.2 : Devriye botu tipleri.

Devriye Botu Tipi	Menzili (km)	Ait Olduğu Bölge
1	8000	Bölge-1
2	100	Bölge-2
3	200	Bölge-2
4	200	Bölge-3
5	300	Bölge-3
6	1000	Bölge-4
7	300	Bölge-4

6.1.2 Verilerin analizi

Uygulama aşamasında Sahil Güvenlik Komutanlığı personeli ile istişare ederek gerekli bazı varsayımları yapıyoruz, Öncelikle Sahil Güvenlik Komutanlığı 7/24 operasyona dayalı olduğundan devriye sırasında kesintisiz çalıştığını varsayıyoruz. İkinci olarak, öklid mesafe fonksiyonuna göre askeri limanların sıcak noktalara olan mesafesini hesaplıyoruz ve askeri liman içinde kat edilen mesafeleri yok sayıyoruz. Verilerin Bölüm 6.1.1’de anlatıldığı üzere bölgelere ve haftalık bazlı verilere dönüştürülmesi ile elde edilen veri setleri öncelikle genel durumu gözlemlemek ve göç olayına ilişkin genel durumu kavramak amacıyla analiz ediyoruz. Bu analiz göç olaylarına ilişkin varsa özel tarihleri, mevsimselliği, önemli politik günlere olan tepkisini gözlemlemek amacıyla önemlidir. Bu doğrultuda öncelikle 1 Ocak 2015 ile 31 Ağustos 2019 arasında Sahil Güvenlik Komutanlığı tarafından müdahale edilen günlük göçmen sayısı analizi yapılır ve bu analiz Şekil 6.3’te paylaşılır.



Şekil 6.3 : Kıta sahanlığı için haftalık bazda müdahale edilen göçmen analizi.

Şekil 6.3 incelendiğinde Temmuz 2015 ve Nisan 2016 arasında diğer dönemlere göre daha fazla mülteci tespiti yapıldığı görülmektedir. Bu duruma farklı bırakış açısı getirmek gerekirse Nisan 2016 itibariyle tespit edilen göçmen sayısı ani azalmaktadır. Burada önemli bir siyasi gelişmeyi hatırlatmakta fayda vardır. Mart 2016'da Brüksel'de Avrupa Birliği (AB) liderleri ile gerçekleştirilen toplantıda mültecilerle ilgili önemli kararlar alınmıştır. Bu kararlardan en önemlisi, Ege'de yakalanan göçmen Suriyeli değilse ülkesine iade edilecek, Suriyeli ise kampa yerleştirilecektir. Bunun karşılığında kampa yerleştirilen her Suriyeli göçmen için Avrupa ülkeleri bir Suriyeli göçmeni ülkesine kabul edecektir. Dolayısıyla Ege'de yakalanan her göçmen Türkiye'deki kampa yerleştirilecek ve dolaylı olarak Batı ülkelerine göç etmesi engellenmiş olacaktır. Bu durumun da bir süre göç olayında kesinti yaşanmasına neden olmuş olabileceği ve grafikteki azalışın nedeni olabileceği değerlendirilmektedir.

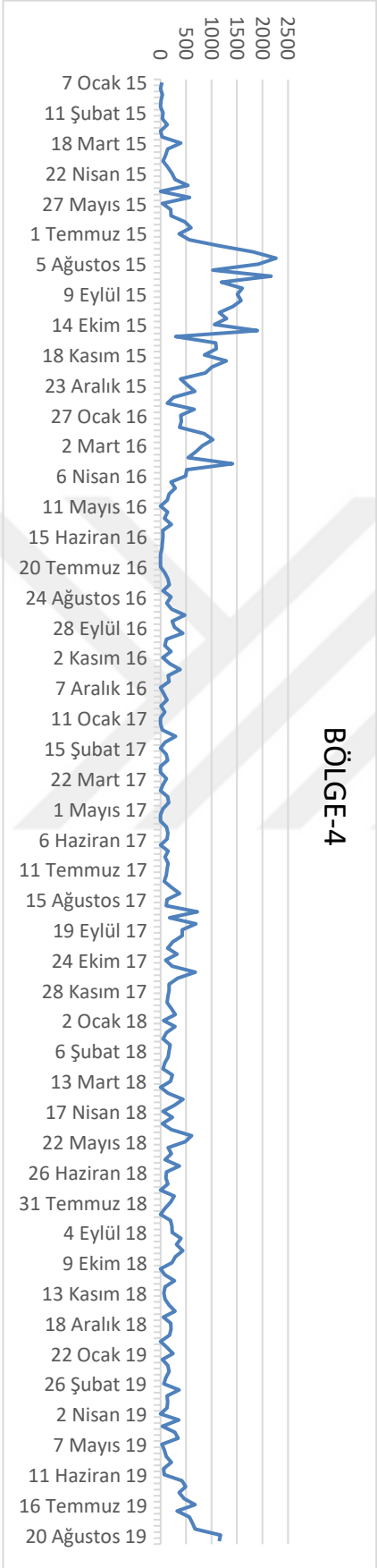
Farklı bir durum olarak Temmuz 2016 döneminde müdahale edilen göçmen sayısının azaldığı görülmektedir. Bu durum için de çeşitli yorumlar yapılabilir. 15 Temmuz 2016'da meydana gelen darbe girişiminin başarısız sonuçlanmasının ardından göç olayının azalmış olabileceği veya göçmen kaçakçılığına yardım eden kaçakçıların bir süre ara vermiş olabileceği değerlendirilmektedir.

Her iki durum da göç olayının siyasi gelişmelerden çok etkilendiğini göstermektedir. Göç kavramının doğasını anlamanın, sağlıklı bir göçmen sayısı tahmini yapmak için önemli olabileceği değerlendirilmektedir.

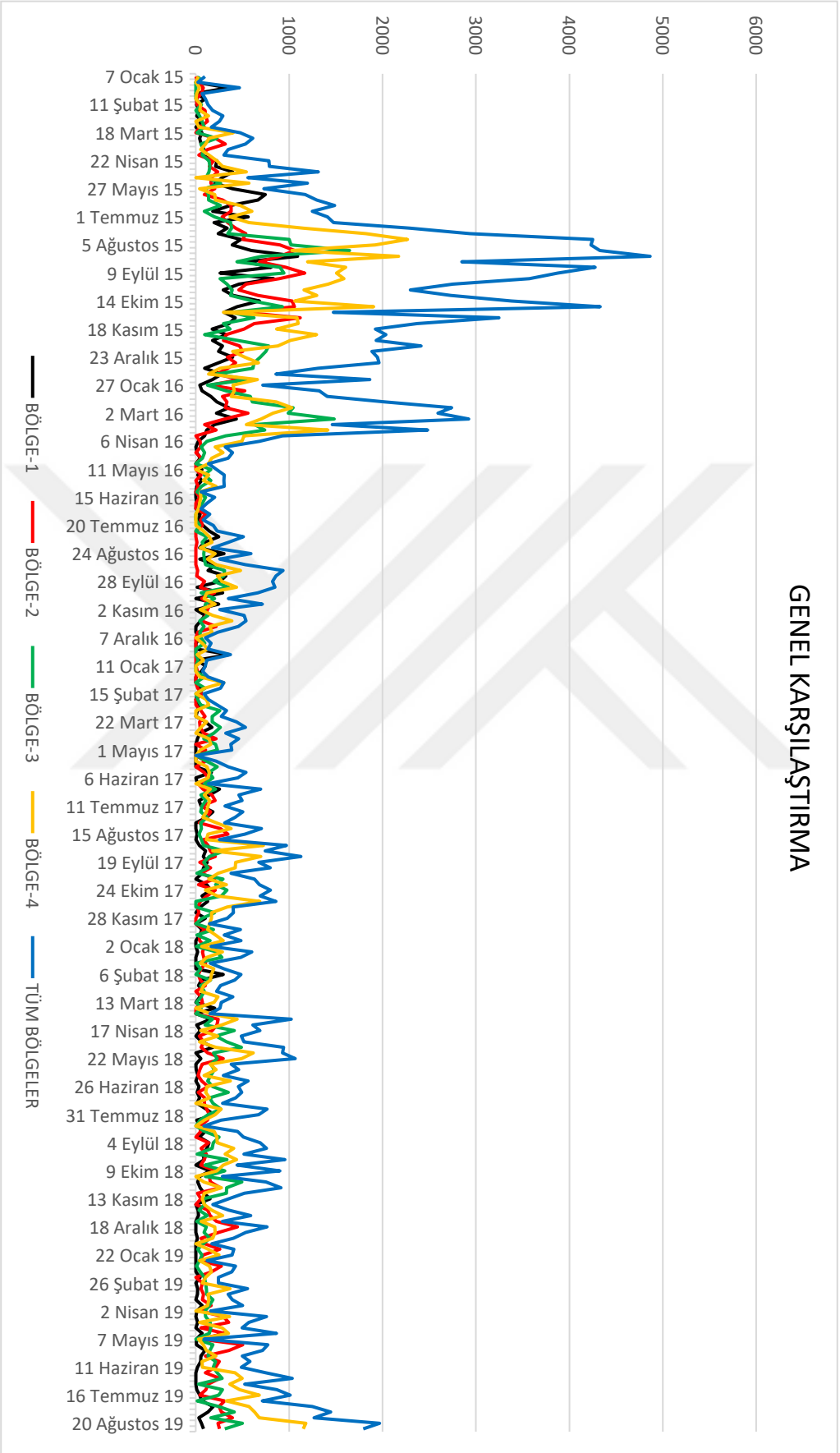
Bölgelere ayrılan veri setleri üzerinde de haftalık bazda müdahale edilen göçmen sayıları Şekil 6.4, Şekil 6.5, Şekil 6.6 ve Şekil 6.7'de paylaşılmıştır.

Son olarak tüm bölgelerin haftalık bazda müdahale edilen göçmen sayıları farklı olsa da grafik izdüşümlerinin benzer yapı gösterdiği görülmektedir. Aynı zamanda bu grafiklerin bölgeden bağımsız olarak tüm veri setine ait grafiklerle de benzer hareket sergilediği görülür. Bu karşılaştırma Şekil 6.8'de verilir.





GENEL KARŞILAŞTIRMA



Şekil 6.8 : Genel karşılaştırma.

Şekil 6.8’de detaylı olarak görüldüğü üzere 2015 yılı Ocak-Nisan ayları arasındaki göçmen yoğunluğu diğer tüm dönemlere göre farklılık sergilemektedir. İlgili yoğunluğun tahminler üzerinde etkisini ortadan kaldırmak amacıyla tahmin çalışmaları 2016 Nisan-2019 Ağustos ayları arasındaki veriler ile elde edilir.

6.1.3 Problemin varsayımları

Uygulama aşamasında Sahil Güvenlik Komutanlığı personeli ile istişare ederek gerekli bazı varsayımları yapıyoruz, Öncelikle Sahil Güvenlik Komutanlığı 7/24 operasyona dayalı olduğundan devriye sırasında kesintisiz çalıştığını varsayıyoruz. İkinci olarak, öklid mesafe fonksiyonuna göre askeri limanların sıcak noktalara olan mesafesini hesaplıyoruz ve askeri liman içinde kat edilen mesafeleri yok sayıyoruz. Üçüncü olarak her botun ait olduğu limana geri dönmek zorunda olduğunu varsayıyoruz. Son olarak her sıcak noktaya en fazla bir kere gidildiğini varsayıyoruz.

6.2 Göçmen sayısı tahmini

Bu çalışma kapsamında Türkiye’de deniz yolu üzerindeki göçmen sayısı tahmin etmek için 5 ayrı bölge (veri seti) üzerinde 8 farklı yöntem kullanılmıştır. Bu yöntemler ARIMA, SES, HES gibi geleneksel yöntemler ve ANFIS, ANN, LSTM ve R-T1FFs gibi alternatif yöntemler olarak gruplandırılabilir.

Ardından, daha sağlam tahmin sonuçları elde etmek için önerilen FCA-MFFs yaklaşımı uygulanır. ARIMA, SES ve HES, zaman serisinin olasılık kısmıyla ilgilenmek için kullanılır. ARIMA, geleneksel olduğu ve genellikle olasılıksal olmayan yöntemler gibi daha karmaşık yaklaşımlardan daha iyi sonuçlar verdiği için seçilmiştir. SES ve HES, ARIMA'dan daha az veri gerektirmesinin yanı sıra, zaman serisinin olasılıklı kısmı ile başa çıkmak amacıyla uygulanır. Çeşitli faktörler nedeniyle göç hareketlerinin karmaşıklığı çok yüksek olduğundan, ANN'nın doğrusal olmayan analiz yapabilme yeteneği, göçmen sayısını tahmin etmede bir avantaj olarak görülmektedir. ANFIS, zaman serisinin olasılıksal olmayan kısmıyla başa çıkmak ve onu eğitmek için yeterli veri olduğunda, ANN'ye göre kanıtlanmış üstünlüğü nedeniyle uygulamaya değer olarak kabul edilir (Atmaca 2001). LSTM ve R-T1FFs, olasılıklı olmadıkları ve tahmin performansları önemli olduğu için seçilir. LSTM yöntemi, verilen verinin özelliklerini daha uzun süre koruduğu ve kendi eğitiminde kullandığı için uygulanmaktadır. R-T1FFs, kural tabanlı olmadığı için uygulama açısından değerli kabul edilir, dolayısıyla uygulaması kolaydır.

Ancak başlangıçta verinin yapısı bilinemeyeceğinden hangi yöntemin daha iyi sonuç vereceği bilinmemektedir. Bu nedenle, FCA-MFFs için bu 7 yöntemi seçerek, verilerin durağanlığı, doğrusallığı veya doğrusal olmaması, az veya yeterli veri, uzun süreli bağımlılıklar veya değil, trendli veya değil gibi tüm durumların ilk etapta ele alınması sağlanır. Ek olarak, FCA-MFFs ile daha düşük RMSE ile daha güvenilir ve sağlam sonuçlar elde edilmesi amaçlanmaktadır. Böylece Sahil Güvenlik Komutanlığı'nın yakalayabileceği göçmenler için güvenilir tahmin sonuçları ile zemin hazırlanmaktadır.

Tüm hesaplamalar, istatistiksel bir programlama dili olan R kullanılarak yapılır. Önerilen yöntemlerin performansı Denklem (6.1) 'de verilen RMSE kullanılarak değerlendirilmiştir.

Daha düşük RMSE değeri, her iki metrik için de daha iyi tahmin doğruluğu anlamına gelir (Tak 2018).

RMSE matematiksel olarak aşağıdaki şekilde ifade edilebilir:

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n (x_t - \hat{x}_t)^2} \quad (6.1)$$

6.2.1 Önerilen tahmin yaklaşımı (FCA-MFFs)

Önerilen Meta Bulanık Fonksiyon Tabanlı Tahmin Kombinasyonu Yaklaşımı (FCA-MFFs)'nin adımları aşağıda verilir:

Adım 1. Problem için mevcut tahmin yöntemlerini belirleyin.

Adım 2. X veri seti, eğitim ($X_{eğitim}$), doğrulama ($X_{doğrulama}$) ve test (X_{test}) olmak üzere üç gruba ayrılır.

- $X = [X_i]$, $i = 1, 2, \dots, n$. Burada n toplam veri sayısıdır.
- $X_{eğitim} = [X_i]$, $i = 1, 2, \dots, neğitim$. Burada $neğitim$, eğitim verilerinin sayısıdır. $X_{eğitim}$, FCA-MFFs yaklaşımının eğitim aşaması için mevcut yöntemlerin eğitiminde kullanılır.
- $X_{doğrulama} = [X_i]$; $i = 1, 2, \dots, ndoğrulama$. Burada $ndoğrulama$, doğrulama verilerinin sayısıdır. $X_{doğrulama}$, en iyi MFF'yi seçmek için kullanılır.

- $X_{test} = [X_i] ; i = 1, 2, \dots, n_{test}$. Burada n_{test} , test verilerinin sayısıdır. X_{test} , FCA-MFF'ler ile tahmin edilecek gözlemlerin gerçek veri kümesidir.

Adım 3. Bir doğrulama veri kümesi $X_{doğrulama}$ için $X_{eğitim}$ kullanarak ilgili mevcut yöntemleri eğitin.

Adım 4. $X_{doğrulama}$ doğrulama veri kümesi için eğitilmiş yöntemleri kullanarak tahmin sonuçlarını elde edin. Tahmin sonuç matrisi, FCA-MFFs yönteminin $Z_{doğrulama}$ adı verilen giriş matrisine girdi oluşturur. $Z_{doğrulama}$, FCA-MFFs yönteminin eğitim veri kümesidir.

$$Z_{doğrulama} = [Z_{ij}] ; i = 1, 2, \dots, p ; j = 1, 2, \dots, ndoğrulama.$$

$$Z_{doğrulama} = \begin{bmatrix} Z_{1,1} & \dots & Z_{1,ndoğrulama} \\ \vdots & \ddots & \vdots \\ Z_{p,1} & \dots & Z_{p,ndoğrulama} \end{bmatrix}$$

Burada Z_{ij}, j . veri noktası için i . yöntemin sonuçlarıdır.

Adım 5. Fonksiyonlarda yöntemlerin ağırlıklarını belirleyin. Giriş matrisi $Z_{doğrulama}$, FCM kullanılarak kümelendir. Her kümedeki üyelik dereceleri, fonksiyonlardaki yöntemlerin ağırlıklarını hesaplamak için kullanılır. Bu durumda, bir küme bir fonksiyonu temsil eder.

Adım 5.1. Bulanık indeks parametresi (m) ve küme sayısı (c) başlatılır. Küme merkezleri (v_i) rastgele başlatılır.

Adım 5.2. Üyelik derecesi Denklem (6.2)'deki gibi hesaplanır.

$$\mu_{ik} = \left[\frac{d(z_i, v_k)^{\frac{2}{m-1}}}{\sum_{j=1}^c d(z_i, v_j)^{\frac{2}{m-1}}} \right] ; i = 1, 2, \dots, c ; k = 1, 2, \dots, ndoğrulama \quad (6.2)$$

$\sum_{i=1}^c \mu_{ik} = 1$ kısıtı altında, eğer $\mu_{ik} < \alpha$ -cut ise sıfır değerini alır. Z giriş

matrisidir, v_i, v_j küme merkezleridir, $d(.)$ Öklid uzaklık fonksiyonu anlamına gelir, c küme sayısıdır ve m Denklem (6.2) ve (6.3)'deki bulanık indeks parametresidir.

Adım 5.3. Yeni küme merkezleri Denklem (6.3)'de hesaplanır.

$$v_i = \frac{\sum_{k=1}^n \mu_{ik}^m z_k}{\sum_{k=1}^n \mu_{ik}^m} \quad (6.3)$$

Adım 5.4. Tekrar sayısına ulaşılan veya iki yineleme arasındaki küme farkı tanımlanan bir eşiğin altına düşene kadar Adım 3 – 5 tekrarlanır.

Adım 6. Denklem (6.5)'de verilen yöntemlerin ağırlıkları kullanılarak Denklem (6.4)'teki MFF'leri elde edin.

$$MFF_i(z) = \sum_{j=1}^m w_{ij} z_j \quad ; i = 1, 2, \dots, c \quad (6.4)$$

$$w_{ij} = \frac{\mu_{ij}}{\sum_{j=1}^m \mu_{ij}} \quad ; i = 1, 2, \dots, c \quad (6.5)$$

Burada, MFF_i , i . meta bulanık fonksiyonunu temsil ederken, μ_{ij} , i . kümedeki j . yöntemin üyelik derecesini, w_{ij} , i . kümedeki j . yöntemin ağırlığını ve c küme sayısını temsil eder.

Adım 7. Farklı m ve c için Adım 5 ve Adım 6'yı tekrarlayın.

Adım 8. Bir fonksiyondaki yöntemlerin en iyi kombinasyonunu seçin. En düşük RMSE olan fonksiyon $Z_{doğrulama}$ yardımıyla MFF_{best} olarak seçilir.

Adım 9. FCA-MFFs algoritması ile tahmin sonuçları elde edin.

Adım 9.1. MFF_{best} , $Z_{doğrulama}$ 'ya göre elde edildiğinden, bu fonksiyon tahmin için kullanılabilir.

Adım 9.2. X_{test} test veri kümesi için $X_{doğrulama}^*$ kullanarak ilgili mevcut yöntemleri eğitin ve tahmin sonuçları matrisini (Z_{test}) elde edin. $X_{doğrulama}^* = [X_i]$, $i = 1, 2, \dots, n_{eğitim} + n_{doğrulama}$ olmak üzere,

$$Z_{test} = [Z_{ij}] \quad ; i = 1, 2, \dots, p \quad ; j = 1, 2, \dots, n_{test}.$$

$$Z_{test} = \begin{bmatrix} Z_{1,1} & \cdots & Z_{1,n_{test}} \\ \vdots & \ddots & \vdots \\ Z_{p,1} & \cdots & Z_{p,n_{test}} \end{bmatrix}$$

Burada Z_{ij} , j . veri noktası için i . yöntemin sonuçlarıdır.

Adım 9.3. FCA-MFFs yönteminin tahmin sonuçları, Denklem (6.6)'daki gibi Z_{test} için MFF_{best} kullanılarak hesaplanır.

$$F = MFF_{best}(Z_{test}) \quad (6.6)$$

Burada F , FCA-MFFs algoritmasının verdiği tahmin sonuçlarıdır.

6.2.1.1 Kıta sahanlığı veri seti üzerinde göçmen sayısı tahmini

İlk hesaplama veri seti olarak Kıta Sahanlığı Bölgesi kullanılır. Türkiye kıta sahanlığı içinde yer alan deniz hattında tespit edilen göçmenlere ilişkin verilerden oluşmaktadır. Bu bölge coğrafi olarak Türkiye'nin güneyi, kuzeyi ve batısındaki ve Türkiye Cumhuriyeti'nin müdahale sınırları içindeki deniz bölgesini ifade eder. İlgili veri seti Nisan 2016 ile Temmuz 2019 arasında haftalık olarak tespit edilen göçmen sayılarından oluşmaktadır. Kıta Sahanlığı Bölgesine ait 155 gözlem verisi bulunmaktadır. Bu verilerin grafiği Şekil 6.9'da sunulur.

FCA-MFFs algoritmasında, Kıta Sahanlığı Bölgesi eğitim veri seti olan $X_{eğitim}$ kullanılarak geleneksel ve alternatif yöntemler eğitilir ve $X_{doğrulama}$ için tahmin sonuçları elde edilir, Bu tahmin sonuçları FCA-MFFs için giriş matrisi olan $X_{eğitim}$ $Z_{doğrulama}$ matrisini oluşturur. Çizelge 6.3'de verilen $Z_{doğrulama}$, FCA-MFFs algoritmasının eğitim aşamasında MFF'leri oluşturmak ve MFF'lerin en iyisini seçmek için kullanılır.

Tüm fonksiyonları oluşturmak için, küme sayısı 2'den 4'e ayarlandığında ve bulanık indeks parametresi 0,5'lik artış oranıyla 1,5'ten 2,5'e ayarlandığında $Z_{doğrulama}$ kullanılarak FCM gerçekleştirilir. FCA-MFFs algoritması, Kıta Sahanlığı Bölgesi veri kümesi için bu koşullar altında en iyi işlevi arar. Tüm iterasyonlar arasında en düşük RMSE değerine sahip fonksiyon, küme sayısı 4 ve bulanık indeks parametresi 2 olduğunda elde edilir. Bu da metotların 4 küme altında toplandığı yani 4 fonksiyon elde ettiğimiz anlamına gelir. Bu fonksiyonlar, biri en düşük RMSE'ye sahip en iyi fonksiyon olmak üzere Denklem (6.7)'de MFF_1 , Denklem (6.8)'de MFF_2 , Denklem (6.9)'da MFF_3 ve Denklem (6.10)'de MFF_4 'tür ve fonksiyonlar Çizelge

6.4'deki ağırlıklara göre oluşturulmuştur. Kümelerde (fonksiyonlarda) metotların ağırlıkları ve fonksiyonların $Z_{doğrulama}$ için RMSE değerleri Çizelge 6.4'de verilmiştir.

$$MFF_1 = 0,026 \times ARIMA + 0,112 \times SES + 0,002 \times HES + 0,088 \times ANFIS + 0,730 \times LSTM + 0,041 \times R-T1FFs \quad (6.7)$$

$$MFF_2 = 0,045 \times ARIMA + 0,078 \times SES + 0,002 \times HES + 0,827 \times ANN + 0,027 \times ANFIS + 0,001 \times LSTM + 0,020 \times R-T1FFs \quad (6.8)$$

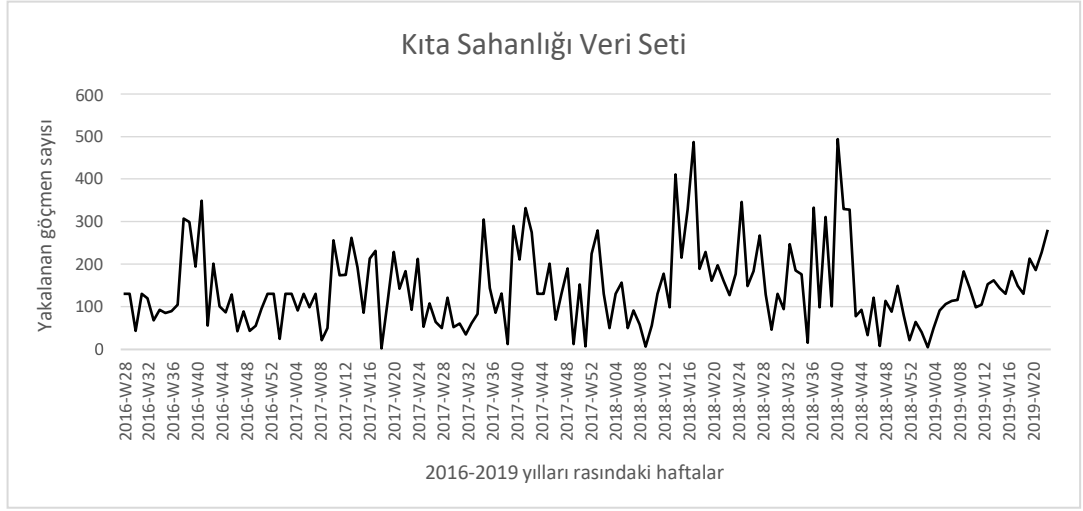
$$MFF_3 = 0,073 \times ARIMA + 0,164 \times SES + 0,659 \times HES + 0,065 \times ANFIS + 0,001 \times LSTM + 0,037 \times R-T1FFs \quad (6.9)$$

$$MFF_4 = 0,256 \times ARIMA + 0,175 \times SES + 0,002 \times HES + 0,262 \times ANFIS + 0,001 \times LSTM + 0,304 \times R-T1FFs \quad (6.10)$$

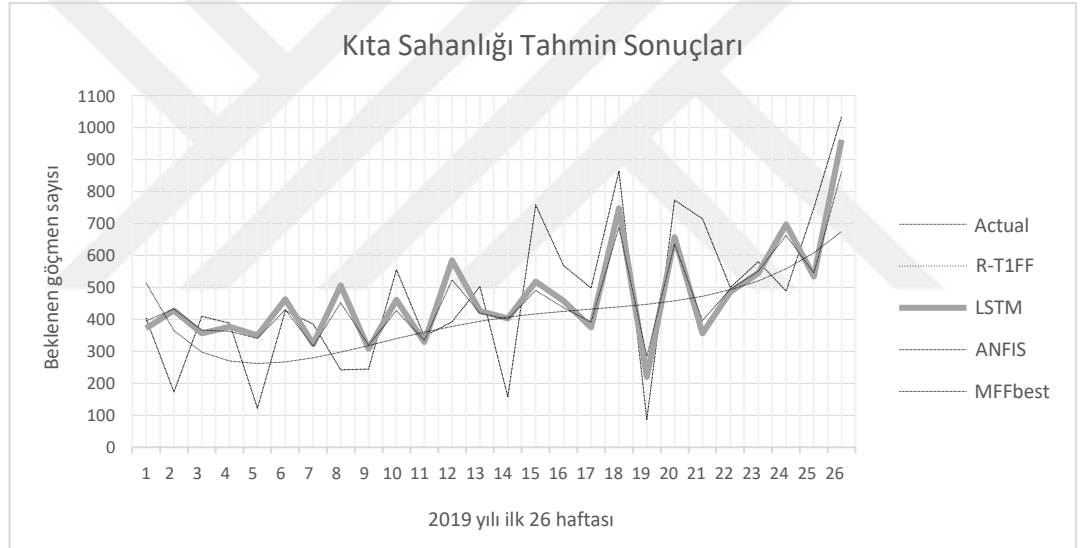
Çizelge 6.4'de MFF_{best} 'in RMSE açısından ilk fonksiyon olduğunu gösterir. Bu nedenle, tahminler Z_{test} için MFF_1 kullanılarak belirlenir. Çizelge 6.5'te verilen Z_{test} 'i elde etmek için 155 veriden ilk 129'u kullanılarak mevcut yöntemler eğitilir ve son 26 gözlem olan test veri seti (X_{test}) tahmin edilir. Z_{test} matrisi ve MFF_1 fonksiyonu kullanılarak MFF_{best} , Denklem (6.11)'deki gibi elde edilir. FCA-MFFs ve diğer mevcut yöntemlerin tahmin sonuçları Çizelge 6.6'da verilir.

$$MFF_{best} = MFF_1(Z_{test}) = 0.026 \times ARIMA + \dots + 0.730 \times LSTM + 0.041 \times R-T1FFs \quad (6.11)$$

Kıta Sahanlıđı Bölgesi veri seti için, LSTM yöntemi, Denklem (6.11) ve Çizelge 6.4'de görüldüđü gibi 0,730 ağırlıkla önerilen FCA-MFFs yöntemine en büyük katkıyı sağlar. LSTM yöntemini sırasıyla 0,026, 0,112, 0,002, 0,088, 0,041 ağırlıkları ile ARIMA, SES, HES, ANFIS, R-T1FFs yöntemleri izlemektedir. Bu veri seti için ANN önerilen yönteme katkıda bulunmaz. Çizelge 6.6'da önerilen yöntemin performansının mevcut yöntemlerle karşılaştırmalı olarak değerlendirilmesi sunulmaktadır. RMSE değerleri açısından en iyi tahmin sonuçları önerilen yöntemden elde edilmektedir. Yöntemlerin tahmin sonuçları Şekil 6.10'da gösterilmektedir.



Őekil 6.9 : Kıta sahanlıđı verileri.



Őekil 6.10 : Tahmin sonularının karŐılaŐtırılması.

Çizelge 6.3 : Kıta sahanlığı $Z_{doğrulama}$ veri seti.

Gözlem No	t	$X_{doğrulama}$	ARIMA	SES	HES	ANN	ANFIS	LSTM	R-TIFFs
104	2018-hafta27	15	34	81	221	62	24	24	39
105	2018-hafta28	39	29	79	154	71	25	53	10
106	2018-hafta29	13	26	76	214	67	31	33	30
107	2018-hafta30	110	28	74	187	69	41	80	38
108	2018-hafta31	195	30	71	167	73	54	128	39
109	2018-hafta32	125	50	73	284	74	70	98	82
110	2018-hafta33	46	74	79	123	31	84	96	74
111	2018-hafta34	103	81	81	100	61	95	90	41
112	2018-hafta35	37	76	79	137	73	101	83	99
113	2018-hafta36	141	79	81	127	72	102	89	118
114	2018-hafta37	119	74	78	161	73	99	87	78
115	2018-hafta38	74	83	82	214	71	92	80	89
116	2018-hafta39	92	88	83	121	73	84	94	69
117	2018-hafta40	60	86	83	192	74	77	76	64
118	2018-hafta41	150	87	83	82	74	69	69	100
119	2018-hafta42	39	84	82	134	72	63	61	90
120	2018-hafta43	19	91	86	144	69	58	58	77
121	2018-hafta44	45	85	83	134	73	55	57	39
122	2018-hafta45	88	77	80	213	71	52	50	22
123	2018-hafta46	152	73	78	111	74	50	59	130
124	2018-hafta47	60	75	79	106	74	49	59	65
125	2018-hafta48	24	84	82	129	73	48	60	51
126	2018-hafta49	29	81	81	76	74	48	49	34
127	2018-hafta50	60	74	78	83	74	48	61	66
128	2018-hafta51	60	69	76	122	74	49	56	83
129	2018-hafta52	60	68	75	113	74	50	51	47

Çizelge 6.4 : Yöntemlerin fonksiyonlardaki ağırlıkları ve $Z_{doğrulama}$ için RMSE değerleri.

Yöntem	MFF_1	MFF_2	MFF_3	MFF_4
ARIMA	0,026	0,045	0,073	0,256
SES	0,112	0,078	0,164	0,175
HES	0,002	0,002	0,659	0,002
ANN	0,000	0,827	0,000	0,000
ANFIS	0,088	0,027	0,065	0,262
LSTM	0,730	0,001	0,001	0,001
R-TIFFs	0,041	0,020	0,037	0,304
RMSE	200,05*	296,88	252,09	235,20

Çizelge 6.5 : Kıta sahanlığı Z_{test} veri seti ve yöntemlerin RMSE değerleri.

X_{test}	ARIMA	SES	HES	ANN	ANFIS	LSTM	R-TIFFs
403	535	402	238	380	514	372	457
173	506	448	674	366	365	428	535
409	436	433	323	339	298	356	450
388	366	342	292	359	270	377	355
122	377	365	351	317	262	350	248
427	373	373	518	349	267	463	378
385	323	286	481	322	280	323	346
242	344	335	304	363	298	506	200
244	351	352	350	327	318	308	407
555	330	314	439	342	340	461	391
346	314	290	530	336	360	329	482
392	360	382	475	340	378	584	234
502	357	370	644	359	394	426	527
157	364	377	746	793	407	403	415
758	390	421	544	590	417	518	397
568	346	329	688	592	425	458	423
497	423	478	600	582	432	375	319
864	450	510	612	583	439	748	809
87	459	505	524	608	447	220	355
773	534	630	437	604	458	658	717
715	451	441	572	630	472	356	787
499	511	557	770	582	493	486	443
580	549	612	685	578	520	544	553
488	540	573	370	539	558	698	607
749	547	575	929	575	608	534	505
1034	536	545	692	576	675	962	585
RMSE	220.02	210.01	240.82	219.93	198.93	160.20	182.69

Çizelge 6.6 : Kıta sahanlığı veri seti için mevcut yöntemler ve FCA-MFFs tahmin sonuçları ve RMSE karşılaştırması.

Gözlem	ARIMA	SES	HES	ANN	ANFIS	LSTM	R-TIFFs	FCA-MFFs
403	535	402	238	380	514	372	403	395
173	506	448	674	366	365	428	173	432
409	436	433	323	339	298	356	409	366
388	366	342	292	359	270	377	388	362
122	377	365	351	317	262	350	122	340
427	373	373	518	349	267	463	427	430
385	323	286	481	322	280	323	385	316
242	344	335	304	363	298	506	242	451
244	351	352	350	327	318	308	244	319
555	330	314	439	342	340	461	555	427
346	314	290	530	336	360	329	346	334
392	360	382	475	340	378	584	392	523
502	357	370	644	359	394	426	502	419
157	364	377	746	793	407	403	157	400
758	390	421	544	590	417	518	758	490
568	346	329	688	592	425	458	568	436
497	423	478	600	582	432	375	497	391
864	450	510	612	583	439	748	864	688
87	459	505	524	608	447	220	87	284
773	534	630	437	604	458	658	773	636
715	451	441	572	630	472	356	715	396
499	511	557	770	582	493	486	499	494
580	549	612	685	578	520	544	580	550
488	540	573	370	539	558	698	488	663
749	547	575	929	575	608	534	749	545
1034	536	545	692	576	675	962	1034	862
RMSE	220.02	210.01	240.82	219.93	198.93	160.20	182.69	159.85*

6.2.1.2 Bölge-1 veri seti üzerinde göçmen sayısı tahmini

Bölge-1, ikinci hesaplama veri seti olarak kullanılır ve Türkiye'nin batı ve güney kıyılarındaki deniz alanından elde edilen ve 35 derece ile 37 derece arasındaki enlemlere denk gelen geçmiş haftalık gözlem değerlerini ifade etmektedir. İlgili veri seti Bölge-1'de Nisan 2016 ile Temmuz 2019 tarihleri arasında deniz hattında haftalık olarak tespit edilen göçmen sayılarından oluşmaktadır. Bölge-1'e ait 155 gözlem verisi bulunmaktadır. Bu verilerin çizgi grafiği Şekil 6.11'de sunulur.

FCA-MFFs algoritmasında, Bölge-1 eğitim veri seti olan $X_{eğitim}$ kullanılarak geleneksel ve alternatif yöntemler eğitilir ve $Z_{doğrulama}$ için tahmin sonuçları elde edilir. Bu tahmin sonuçları FCA-MFF'ler için giriş matrisi olan $Z_{doğrulama}$ 'ı oluşturur. Çizelge 6.7'de verilen $Z_{doğrulama}$, FCA-MFFs algoritmasının eğitim aşamasında MFF'leri oluşturmak ve MFF'lerin en iyisini seçmek için kullanılır.

Tüm fonksiyonları oluşturmak için, küme sayısı 2'den 4'e ayarlandığında ve bulanık indeks parametresi 0,5'lik artış oranıyla 1,5'ten 2,5'e ayarlandığında $Z_{doğrulama}$ kullanılarak FCM gerçekleştirilir. FCA-MFFs algoritması, Bölge-1 veri kümesi için bu koşullar altında en iyi fonksiyonu arar. Tüm iterasyonlar arasında en düşük RMSE değerine sahip fonksiyon, küme sayısı 3 ve bulanık indeks parametresi 2 olduğunda elde edilir. Bu, yöntemlerin 3 küme altında toplandığı anlamına gelir, yani 3 fonksiyon elde ettik. Bu işlevler içlerinden biri en düşük RMSE değerine sahip MFF_{best} olmak üzere sırasıyla MFF_1 , MFF_2 ve MFF_3 'tür. Kümelerde (fonksiyonlarda) metotların ağırlıkları ve fonksiyonların $Z_{doğrulama}$ için RMSE değerleri Çizelge 6.8'de verilir.

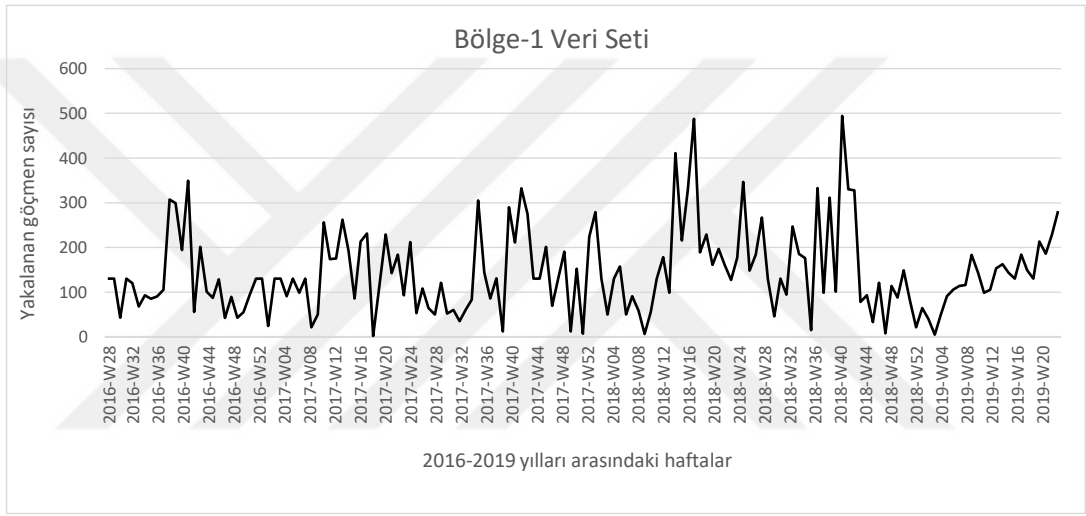
Çizelge 6.8, MFF_{best} 'in RMSE açısından ilk fonksiyon olduğunu ortaya koymaktadır. Bu nedenle, Z_{test} için MFF_1 kullanılarak tahminler elde edilir. Çizelge 6.9'da verilen Z_{test} 'i elde etmek için 155 veriden ilk 129'u kullanılarak mevcut yöntemler eğitilir ve son 26 gözlem olan test veri seti (X_{test}) tahmin edilir. Z_{test} matrisi ve MFF_1 fonksiyonu kullanılarak, MFF_{best} Denklem (6.12)'deki gibi elde edilir. FCA-MFFs verdiği mevcut yöntemler için tahmin sonuçları Çizelge 6.10'da verilir.

$$MFF_{best} = MFF_1(Z_{test}) = 0.085 \times \text{ARIMA} + \dots + 0.334 \times \text{LSTM} + 0.293 \times \text{R-T1FFs}$$

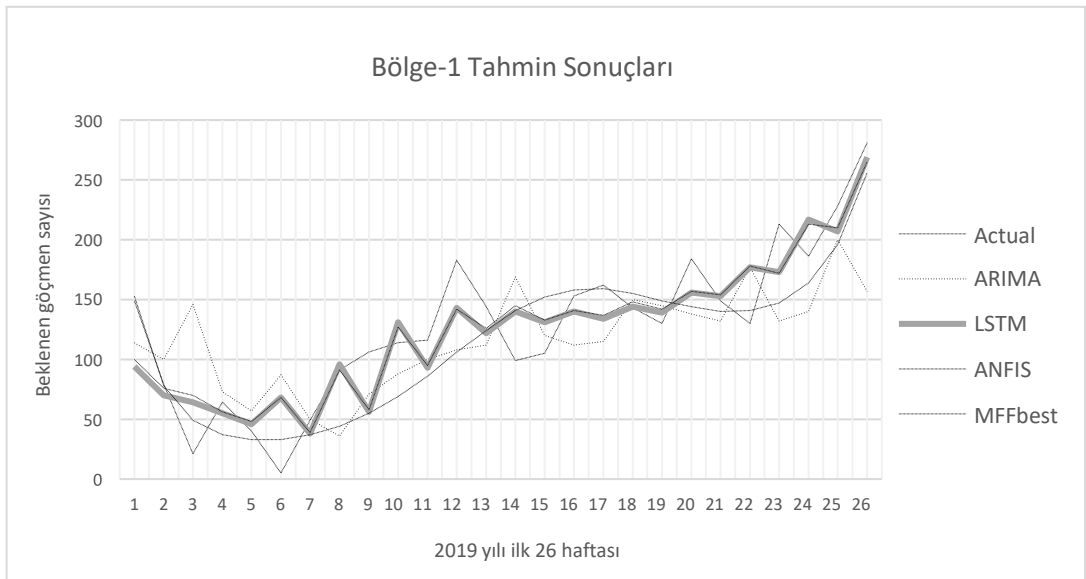
(6.12)

Bölge-1 veri seti için, LSTM yöntemi, Denklem (6.12)'de görüldüğü gibi, 0,334 ağırlıkla önerilen FCA-MFFs yöntemine en büyük katkıyı sağlar. LSTM yöntemini sırasıyla 0,085, 0,005, 0,007, 0,276, 0,293 ağırlıkları ile ARIMA, SES, ANN, ANFIS, R-T1FFs yöntemleri izlemektedir. Bu veri seti için HES, önerilen yönteme katkıda bulunmaz.

Önerilen yöntemin performansını mevcut yöntemlerle değerlendirmek için Çizelge 6.10'da verilmiştir. RMSE değerleri açısından en iyi tahmin sonuçlarının önerilen yöntemden elde edildiği açıktır. Yöntemlerin tahmin sonuçları Şekil 6.12'de gösterilmektedir.



Şekil 6.11 : Bölge-1 verileri.



Şekil 6.12 : Tahmin sonuçlarının karşılaştırılması.

Çizelge 6.7 : Bölge-1 için $Z_{doğrulama}$ veri seti.

Gözlem No	$X_{doğrulama}$	ARIMA	SES	HES	ANN	ANFIS	LSTM	R-T1FFs
104	15	34	81	221	62	24	24	39
105	39	29	79	154	71	25	53	10
106	13	26	76	214	67	31	33	30
107	110	28	74	187	69	41	80	38
108	195	30	71	167	73	54	128	39
109	125	50	73	284	74	70	98	82
110	46	74	79	123	31	84	96	74
111	103	81	81	100	61	95	90	41
112	37	76	79	137	73	101	83	99
113	141	79	81	127	72	102	89	118
114	119	74	78	161	73	99	87	78
115	74	83	82	214	71	92	80	89
116	92	88	83	121	73	84	94	69
117	60	86	83	192	74	77	76	64
118	150	87	83	82	74	69	69	100
119	39	84	82	134	72	63	61	90
120	19	91	86	144	69	58	58	77
121	45	85	83	134	73	55	57	39
122	88	77	80	213	71	52	50	22
123	152	73	78	111	74	50	59	130
124	60	75	79	106	74	49	59	65
125	24	84	82	129	73	48	60	51
126	29	81	81	76	74	48	49	34
127	60	74	78	83	74	48	61	66
128	60	69	76	122	74	49	56	83
129	60	68	75	113	74	50	51	47

Çizelge 6.8 : Yöntemlerin fonksiyonlardaki ağırlıkları ve $Z_{doğrulama}$ için RMSE değerleri.

Yöntem	MFF_1	MFF_2	MFF_3
ARIMA	0,085	0,000	0,243
SES	0,005	0,000	0,318
HES	0,000	0,998	0,000
ANN	0,007	0,000	0,316
ANFIS	0,276	0,000	0,011
LSTM	0,334	0,001	0,064
R-T1FFs	0,293	0,001	0,048
RMSE	42,16*	100,05	47,26

Çizelge 6.9 : Bölge-1 için Z_{test} veri seti ve yöntemlerin RMSE değerleri.

X_{test}	ARIMA	SES	HES	ANN	ANFIS	LSTM	R-TIFFs
17	60	46	73	64	38	50	32
60	47	46	76	66	43	58	63
60	44	45	80	79	48	46	27
60	45	46	81	80	52	52	61
60	52	46	76	68	54	33	32
60	50	47	303	52	55	49	58
60	54	48	81	46	54	38	56
46	53	48	53	38	52	47	57
60	55	49	95	50	49	42	56
20	53	49	64	53	45	47	49
18	53	49	41	50	42	46	55
60	51	48	198	50	38	43	33
68	49	46	36	54	35	48	39
11	49	47	189	53	32	33	29
60	49	48	46	147	30	38	39
21	50	46	81	64	29	40	35
60	46	47	48	61	28	48	58
68	50	46	94	98	27	29	22
6	45	46	189	90	27	52	48
7	50	47	33	61	28	25	9
100	46	46	69	74	29	55	94
53	48	44	59	80	31	41	8
51	44	46	58	185	33	40	38
18	51	47	67	63	36	32	24
60	44	47	103	60	41	60	78
60	50	46	128	74	46	43	35
RMSE	25.01	23.75	80.42	43.65	24.56	22.72	23.29

Çizelge 6.10 : Bölge-1 veri seti için mevcut yöntemler ve FCA-MFFs tahmin sonuçları ve RMSE karşılaştırması.

Gözlem	ARIMA	SES	HES	ANN	ANFIS	LSTM	R-TIFFs	FCA-MFFs
17	60	46	73	64	38	50	32	42
60	47	46	76	66	43	58	63	53
60	44	45	80	79	48	46	27	41
60	45	46	81	80	52	52	61	54
60	52	46	76	68	54	33	32	42
60	50	47	303	52	55	49	58	54
60	54	48	81	46	54	38	56	50
46	53	48	53	38	52	47	57	52
60	55	49	95	50	49	42	56	50
20	53	49	64	53	45	47	49	48
18	53	49	41	50	42	46	55	48
60	51	48	198	50	38	43	33	39
68	49	46	36	54	35	48	39	41
11	49	47	189	53	32	33	29	33
60	49	48	46	147	30	38	39	38
21	50	46	81	64	29	40	35	36
60	46	47	48	61	28	48	58	44
68	50	46	94	98	27	29	22	29
6	45	46	189	90	27	52	48	42
7	50	47	33	61	28	25	9	24
100	46	46	69	74	29	55	94	57
53	48	44	59	80	31	41	8	29
51	44	46	58	185	33	40	38	39
18	51	47	67	63	36	32	24	33
60	44	47	103	60	41	60	78	58
60	50	46	128	74	46	43	35	42
RMSE	25.01	23.75	80.42	43.65	24.56	22.72	23.29	21.59*

6.2.1.3 Bölge-2 veri seti üzerinde göçmen sayısı tahmini

Bölge-2, üçüncü hesaplama veri seti olarak kullanılır ve 37 derece ile 38 derece arasındaki enlemlere denk gelen Türkiye'nin batı kıyısındaki deniz alanından elde edilen geçmiş haftalık gözlem değerlerini ifade eder. İlgili veri seti Bölge-2'de Nisan 2016 ile Temmuz 2019 tarihleri arasında deniz hattında haftalık bazda tespit edilen göçmen sayılarından oluşmaktadır. Bölge-2'ye ait 155 adet gözlem verisi bulunmaktadır. Bu verilerin hat grafiği Şekil 6.13'de sunulmaktadır.

FCA-MFFs algoritmasında, Bölge-2 eğitim veri seti olan $X_{eğitim}$ kullanılarak geleneksel ve alternatif yöntemler eğitilir ve $X_{doğrulama}$ için tahmin sonuçları elde edilir. Bu tahmin sonuçları FCA-MFF'ler için giriş matrisi olan $Z_{doğrulama}$ 'ı oluşturur. Çizelge 6.11'de verilen $Z_{doğrulama}$ FCA-MFFs algoritmasının eğitim aşamasında MFF'leri oluşturmak ve MFF'lerin en iyisini seçmek için kullanılır.

Tüm fonksiyonları oluşturmak için, küme sayısı 2'den 4'e ayarlandığında ve bulanık indeks parametresi 0,5'lik artış oranıyla 1,5'ten 2,5'e ayarlandığında $Z_{doğrulama}$ kullanılarak FCM gerçekleştirilir. FCA-MFFs algoritması, Bölge-2 veri kümesi için bu koşullar altında en iyi işlevi arar. Tüm iterasyonlar arasında en düşük RMSE değerine sahip fonksiyon, küme sayısı 4 ve bulanık indeks parametresi 2 olduğunda elde edilir. Bu da metotların 4 küme altında toplandığı yani 4 fonksiyon elde ettiğimiz anlamına gelir. Bu işlevler sırasıyla MFF_1 , MFF_2 , MFF_3 , ve MFF_4 'tür ve bunlardan biri en düşük RMSE değerine sahip MFF_{best} 'tir. Denklem ile elde edilen yöntemlerin fonksiyonlardaki ağırlıkları. ve $Z_{eğitim}$ için fonksiyonların RMSE değerleri Çizelge 6.12'de verilir.

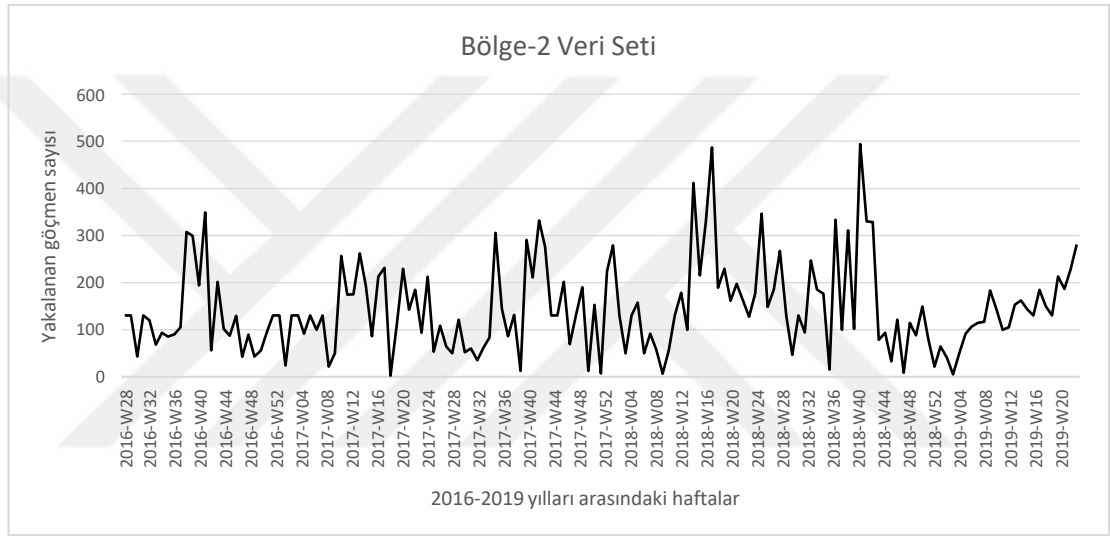
Çizelge 6.12, en iyi MFF'nin (MFF_{best}) RMSE açısından ikinci fonksiyon olduğunu ortaya koymaktadır. Bu nedenle, tahminler $MFF_{best}=MFF_2$ kullanılarak hesaplanır. Çizelge 6.13'de verilen Z_{test} 'i elde etmek için 155 veriden ilk 129'u kullanılarak mevcut yöntemler eğitilir ve son 26 gözlem olan test veri seti (X_{test}) tahmin edilir.

Z_{test} matrisi ve MFF_2 fonksiyonu kullanılarak MFF_{best} , Denklem 6.13'deki gibi elde edilir. ve FCA-MFFs ve diğer mevcut yöntemler için tahmin sonuçları Çizelge 6.14'de verilir.

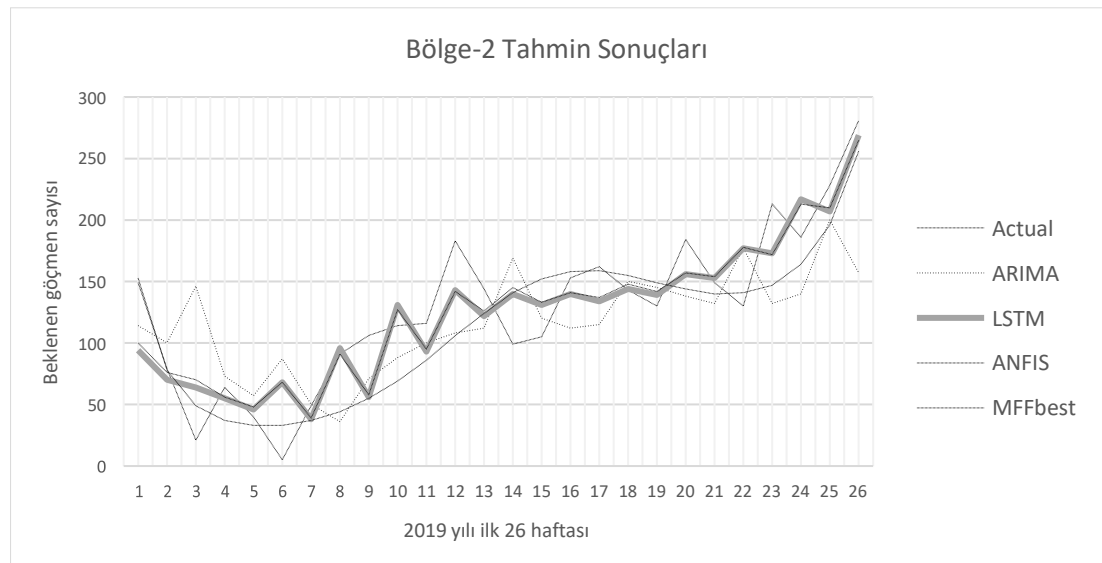
$$MFF_{best} = MFF_2(Z_{test}) = 0.01 \times ANN + 0.009 \times ANFIS + 0.971 \times LSTM(6.13)$$

Bölge-2 veri seti için, LSTM yöntemi, Denklem 6.13'de görüldüğü gibi 0,971 ağırlıkla önerilen FCA-MFFs yöntemine en büyük katkıyı sağlar. LSTM yöntemini sırasıyla 0,010, 0,009, 0,010 ağırlıkları ile ANN, ANFIS, R-T1FFs yöntemleri takip etmektedir. Bu veri seti için SES ve HES önerilen yöntemlere katkı sağlamamaktadır.

Önerilen yöntemin mevcut yöntemlerle performansını değerlendirmek için Çizelge 6.14'de verilmiştir. RMSE değerleri açısından en iyi tahmin sonuçlarının önerilen yöntemden elde edildiği açıktır. Yöntemlerin tahmin sonuçları Şekil 6.14'de gösterilmektedir.



Şekil 6.13 : Bölge-2 verileri.



Şekil 6.14 : Tahmin sonuçlarının karşılaştırılması.

Çizelge 6.11 : Bölge-2 için $Z_{doğrulama}$ veri seti.

Gözlem No	$X_{doğrulama}$	ARIMA	SES	HES	ANN	ANFIS	LSTM	R-TIFFs
104	27	146	68	123	74	66	118	113
105	93	144	78	90	102	81	133	46
106	90	130	70	112	76	90	122	57
107	116	136	73	61	76	93	94	57
108	143	137	76	59	85	91	115	91
109	124	141	82	117	76	88	64	88
110	41	145	92	93	76	85	59	87
111	52	143	97	85	76	82	72	93
112	99	132	88	111	76	80	55	86
113	136	131	82	106	76	80	106	91
114	58	137	85	93	76	81	89	104
115	113	144	93	92	106	83	157	86
116	92	135	87	84	76	87	106	70
117	51	139	91	102	106	92	139	79
118	214	138	92	145	76	97	128	95
119	144	132	85	170	94	103	101	108
120	154	152	105	95	76	109	116	101
121	263	148	111	132	76	115	107	132
122	16	147	118	96	106	120	91	128
123	54	162	141	103	101	124	68	127
124	99	133	121	126	92	127	128	145
125	127	130	111	147	76	129	93	100
126	152	137	109	65	76	129	190	138
127	219	142	112	59	76	128	165	115
128	445	147	118	82	85	127	251	90
129	242	156	134	112	103	124	172	167

Çizelge 6.12 : Yöntemlerin fonksiyonlardaki ağırlıkları ve $Z_{doğrulama}$ için RMSE değerleri.

Yöntem	MFF_1	MFF_2	MFF_3	MFF_4
ARIMA	0,000	0,000	0,974	0,000
SES	0,002	0,000	0,000	0,264
HES	0,854	0,000	0,000	0,000
ANN	0,091	0,010	0,011	0,230
ANFIS	0,010	0,009	0,003	0,260
LSTM	0,000	0,971	0,000	0,000
R-TIFFs	0,043	0,010	0,011	0,246
RMSE	96,73	68,26*	89,28	92,14

Çizelge 6.13 : Bölge-2 için Z_{test} veri seti ve yöntemlerin RMSE değerleri.

X_{test}	ARIMA	SES	HES	ANN	ANFIS	LSTM	R-TIFFs
56	168	128	102	33	155	96	224
94	165	151	48	34	124	96	220
256	173	132	49	39	111	144	167
77	172	124	64	27	106	138	121
43	164	151	49	41	105	153	85
274	172	136	46	42	105	147	121
168	174	117	60	44	103	120	135
99	164	149	106	32	101	110	87
71	168	153	36	54	98	91	140
55	171	142	34	43	95	73	114
82	173	127	85	34	93	78	77
70	173	113	129	54	93	62	60
163	172	106	70	54	94	108	52
47	173	99	74	39	98	92	65
293	169	112	112	32	105	187	73
353	174	99	245	22	116	212	99
54	163	138	30	41	132	170	122
304	160	182	135	41	151	465	196
52	173	156	31	43	174	78	216
502	162	186	138	22	198	427	114
354	174	159	35	3	220	149	205
102	153	229	125	10	232	95	163
252	160	254	178	26	229	313	234
213	171	223	65	51	206	172	254
101	165	229	117	43	164	110	152
257	166	226	212	23	114	277	149
RMSE	122.22	122.07	133.58	184.7	118.92	83	131.59

Çizelge 6.14 : Bölge-2 veri seti için mevcut yöntemler ve FCA-MFFs tahmin sonuçları ve RMSE karşılaştırması.

Gözlem	ARIMA	SES	HES	ANN	ANFIS	LSTM	R-TIFFs	FCA-MFFs
56	168	128	102	33	155	96	224	97
94	165	151	48	34	124	96	220	97
256	173	132	49	39	111	144	167	143
77	172	124	64	27	106	138	121	136
43	164	151	49	41	105	153	85	151
274	172	136	46	42	105	147	121	146
168	174	117	60	44	103	120	135	120
99	164	149	106	32	101	110	87	109
71	168	153	36	54	98	91	140	92
55	171	142	34	43	95	73	114	74
82	173	127	85	34	93	78	77	78
70	173	113	129	54	93	62	60	63
163	172	106	70	54	94	108	52	107
47	173	99	74	39	98	92	65	91
293	169	112	112	32	105	187	73	184
353	174	99	245	22	116	212	99	208
54	163	138	30	41	132	170	122	168
304	160	182	135	41	151	465	196	456
52	173	156	31	43	174	78	216	80
502	162	186	138	22	198	427	114	418
354	174	159	35	3	220	149	205	149
102	153	229	125	10	232	95	163	96
252	160	254	178	26	229	313	234	309
213	171	223	65	51	206	172	254	172
101	165	229	117	43	164	110	152	110
257	166	226	212	23	114	277	149	271
RMSE	122.22	122.07	133.58	184.7	118.92	83	131.59	82.84*

6.2.1.4 Bölge-3 veri seti üzerinde göçmen sayısı tahmini

Bölge-3, dördüncü hesaplama veri seti olarak kullanılır ve Türkiye'nin batı kıyısındaki deniz alanından elde edilen ve 38 derece ile 39 derece arasındaki enlemlere denk gelen geçmiş haftalık gözlem değerlerini ifade eder. İlgili veri seti, Bölge-3'te Nisan 2016 ile Temmuz 2019 tarihleri arasında deniz hattında haftalık bazda tespit edilen göçmen sayılarından oluşmaktadır. Bölge-3'e ait 155 adet gözlem verisi bulunmaktadır. Bu verilerin hat grafiği Şekil 6.15'de sunulmaktadır.

FCA-MFFs algoritmasında, Bölge-3 eğitim veri seti olan $X_{eğitim}$ kullanılarak geleneksel ve alternatif yöntemler eğitilir ve $Z_{doğrulama}$ için tahmin sonuçları elde edilir, Bu tahmin sonuçları FCA-MFF'ler için giriş matrisi olan $Z_{doğrulama}$ 'ı oluşturur. Çizelge 6.15'te verilen $Z_{doğrulama}$, FCA-MFFs algoritmasının eğitim aşamasında MFF'leri oluşturmak ve MFF'lerin en iyisini seçmek için kullanılır.

Tüm fonksiyonları oluşturmak için, küme sayısı 2'den 4'e ayarlandığında ve bulanık indeks parametresi 0,5'lik artış oranıyla 1,5'ten 2,5'e ayarlandığında $Z_{doğrulama}$ kullanılarak FCM gerçekleştirilir. FCA-MFFs algoritması, Bölge-3 veri kümesi için bu koşullar altında en iyi fonksiyonu arar. Tüm iterasyonlar arasında en düşük RMSE değerine sahip fonksiyon, küme sayısı 4 ve bulanık indeks parametresi 2 olduğunda elde edilir. Bu da metotların 4 küme altında toplandığı yani 4 fonksiyon elde ettiğimiz anlamına gelir. Bu işlevler sırasıyla MFF_1 , MFF_2 , MFF_3 , ve MFF_4 'tür ve bunlardan biri en düşük RMSE değerine sahip MFF_{best} 'tir. Yöntemlerin fonksiyonlardaki ağırlıkları, ve $Z_{doğrulama}$ için fonksiyonların RMSE değerleri Çizelge 6.16'da verilmiştir.

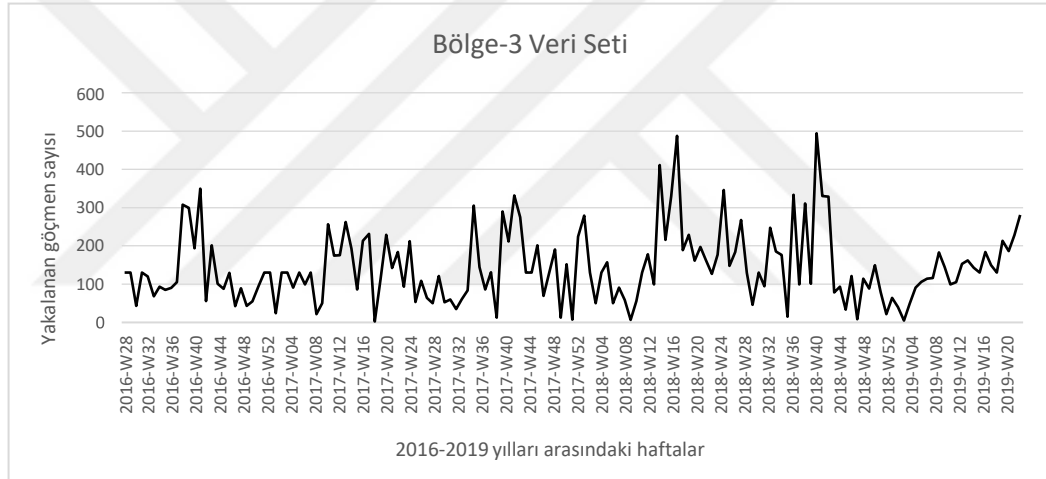
Çizelge 6.16, en iyi MFF'nin (MFF_{best}) RMSE açısından ikinci fonksiyon olduğunu ortaya koymaktadır. Bu nedenle, Z_{test} için MFF_2 kullanılarak tahminler elde edilir. Çizelge 6.17'de verilen Z_{test} 'i elde etmek için 155 veriden ilk 129'u kullanılarak mevcut yöntemler eğitilir ve son 26 gözlem olan test veri seti (X_{test}) tahmin edilir.

Z_{test} matrisi ve MFF_2 fonksiyonu kullanılarak MFF_{best} , Denklem (6.14)'deki gibi elde edilir. FCA-MFFs ve diğer mevcut yöntemler için tahmin sonuçları Çizelge 6.18'de verilir.

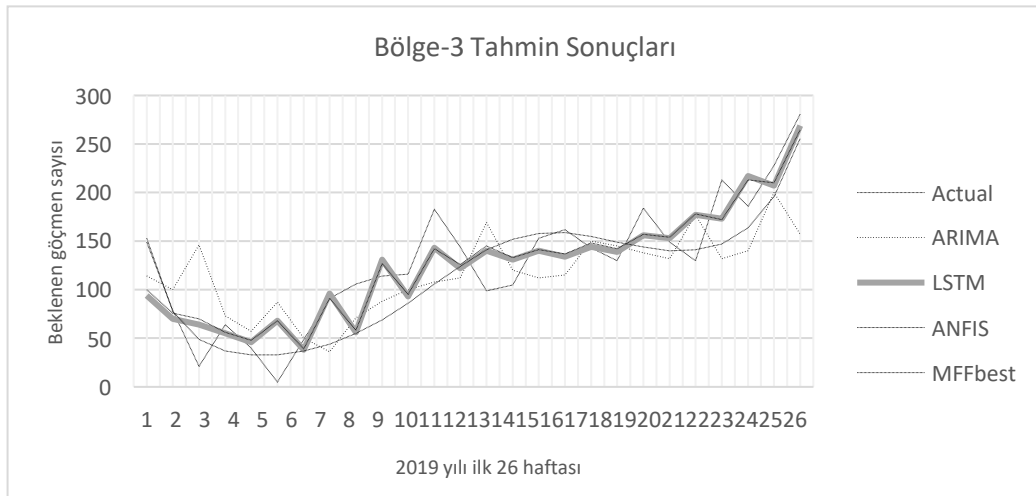
$$MFF_{best} = MFF_2(Z_{test}) = 0.062 \times ARIMA + \dots + 0.879 \times LSTM + 0.019 \times R-T1FFs \quad (6.14)$$

Bölge-3 veri seti için, LSTM yöntemi, Denklem (6.14)'te görüldüğü gibi, 0,879 ağırlıkla önerilen FCA-MFFs yöntemine en büyük katkıyı sağlar. LSTM yöntemini sırasıyla 0,062, 0,001, 0,005, 0,052, 0,019 ağırlıklarla ARIMA, SES, HES, ANFIS, R-T1FFs yöntemleri izlemektedir. Bu veri seti için ANN önerilen yönteme katkıda bulunmaz.

Önerilen yöntemin mevcut yöntemlerle performansını değerlendirmek için Çizelge 6.18 verilmiştir. RMSE değerleri açısından en iyi tahmin sonuçlarının önerilen yöntemden elde edildiği açıktır. Yöntemlerin tahmin sonuçları Şekil 6.16'da gösterilmektedir.



Şekil 6.15 : Bölge-3 verileri.



Şekil 6.16 : Tahmin sonuçlarının karşılaştırılması.

Çizelge 6.15 : Bölge-3 için $Z_{doğrulama}$ veri seti.

Gözlem No	$X_{doğrulama}$	ARIMA	SES	HES	ANN	ANFIS	LSTM	R-T1FFs
104	346	152	185	74	53	234	160	141
105	148	176	184	166	112	226	148	214
106	184	174	182	213	125	200	169	196
107	267	194	181	146	84	170	176	107
108	129	96	180	76	114	144	144	304
109	46	241	178	131	118	126	183	156
110	130	121	177	134	129	116	130	71
111	94	94	175	86	122	112	158	118
112	247	169	174	93	115	113	174	163
113	185	127	173	134	137	119	145	76
114	176	166	171	104	19	128	189	154
115	15	200	170	102	119	139	21	106
116	333	107	168	155	68	150	306	116
117	99	144	167	170	127	160	93	135
118	311	164	166	181	146	167	312	224
119	101	212	164	316	131	170	108	76
120	494	92	163	260	150	167	470	270
121	330	250	161	304	156	160	131	200
122	328	141	160	333	25	148	330	414
123	78	336	159	179	116	135	121	155
124	93	81	157	225	15	120	88	245
125	33	201	156	148	120	105	111	131
126	121	69	155	185	70	92	112	168
127	8	151	153	170	63	80	109	86
128	114	119	152	123	126	71	121	71
129	88	125	150	172	17	63	82	35

Çizelge 6.16 : Yöntemlerin fonksiyonlardaki ağırlıkları ve $Z_{doğrulama}$ için RMSE değerleri.

Yöntem	MFF_1	MFF_2	MFF_3	MFF_4
ARIMA	0,123	0,062	0,227	0,116
SES	0,041	0,001	0,353	0,046
HES	0,004	0,005	0,006	0,592
ANN	0,635	0,000	0,003	0,002
ANFIS	0,102	0,052	0,284	0,066
LSTM	0,003	0,879	0,003	0,004
R-T1FFs	0,093	0,019	0,124	0,175
RMSE	130,07	72,01*	115,12	111,80

Çizelge 6.17 : Bölge-3 için Z_{test} veri seti ve yöntemlerin RMSE değerleri.

X_{test}	ARIMA	SES	HES	ANN	ANFIS	LSTM	R-TIFFs
149	114	76	127	120	153	94	88
79	100	79	242	69	78	70	146
21	146	82	179	41	49	64	80
64	73	86	173	98	37	55	20
40	57	89	132	98	33	46	65
5	87	92	10	95	33	68	44
49	50	96	48	95	37	38	9
91	36	99	174	98	44	96	54
106	71	102	72	98	55	56	95
114	88	106	97	98	69	131	108
116	100	109	190	121	86	93	109
183	108	112	216	99	106	143	116
144	112	115	224	95	124	122	181
99	169	119	313	94	141	140	142
105	120	122	246	98	152	131	99
153	112	125	157	150	158	140	104
162	115	129	168	116	159	134	151
143	150	132	187	98	155	144	160
130	145	135	73	90	149	139	141
184	138	139	165	81	144	156	128
149	132	142	178	41	140	153	182
130	177	145	191	97	141	177	147
213	132	149	232	41	147	173	129
186	140	152	142	97	164	217	211
228	200	155	255	95	196	207	185
281	157	159	106	94	256	269	230
RMSE	38.06	40.63	88.38	74.68	33.61	30.47	42.18

Çizelge 6.18 : Bölge-3 veri seti için mevcut yöntemler ve FCA-MFFs tahmin sonuçları ve RMSE karşılaştırması.

Gözlem	ARIMA	SES	HES	ANN	ANFIS	LSTM	R-TIFFs	FCA-MFFs
149	114	76	127	120	153	94	88	100
79	100	79	242	69	78	70	146	76
21	146	82	179	41	49	64	80	70
64	73	86	173	98	37	55	20	56
40	57	89	132	98	33	46	65	48
5	87	92	10	95	33	68	44	68
49	50	96	48	95	37	38	9	39
91	36	99	174	98	44	96	54	91
106	71	102	72	98	55	56	95	58
114	88	106	97	98	69	131	108	127
116	100	109	190	121	86	93	109	95
183	108	112	216	99	106	143	116	142
144	112	115	224	95	124	122	181	126
99	169	119	313	94	141	140	142	145
105	120	122	246	98	152	131	99	133
153	112	125	157	150	158	140	104	141
162	115	129	168	116	159	134	151	137
143	150	132	187	98	155	144	160	148
130	145	135	73	90	149	139	141	142
184	138	139	165	81	144	156	128	157
149	132	142	178	41	140	153	182	154
130	177	145	191	97	141	177	147	178
213	132	149	232	41	147	173	129	172
186	140	152	142	97	164	217	211	213
228	200	155	255	95	196	207	185	210
281	157	159	106	94	256	269	230	265
RMSE	38.06	40.63	88.38	74.68	33.61	30.47	42.18	30.26*

6.2.1.5 Bölge-4 veri seti üzerinde göçmen sayısı tahmini

Bölge-4, son hesaplama veri seti olarak kullanılır ve Türkiye'nin batı ve kuzey kıyılarındaki deniz alanından elde edilen ve 39 derece ile 41 derece arasındaki enlemlere denk gelen geçmiş haftalık gözlem değerlerini ifade eder. İlgili veri seti Nisan 2016 ile Temmuz 2019 tarihleri arasında Bölge-4'te deniz hattında haftalık olarak tespit edilen göçmen sayılarından oluşmaktadır. Bölge-4'e ait 155 adet gözlem verisi bulunmaktadır. Bu verilerin çizgi grafiği Şekil 6.17'de sunulmuştur.

FCA-MFFs algoritmasında, Bölge-4 eğitim veri seti olan $X_{eğitim}$ kullanılarak geleneksel ve alternatif yöntemler eğitilir ve $Z_{doğrulama}$ için tahmin sonuçları elde edilir. Bu tahmin sonuçları FCA-MFFs için giriş matrisi olan $Z_{doğrulama}$ 'ı oluşturur. Çizelge 6.19'da verilen $Z_{doğrulama}$, FCA-MFFs algoritmasının eğitim aşamasında MFF'leri oluşturmak ve MFF'lerin en iyisini seçmek için kullanılır.

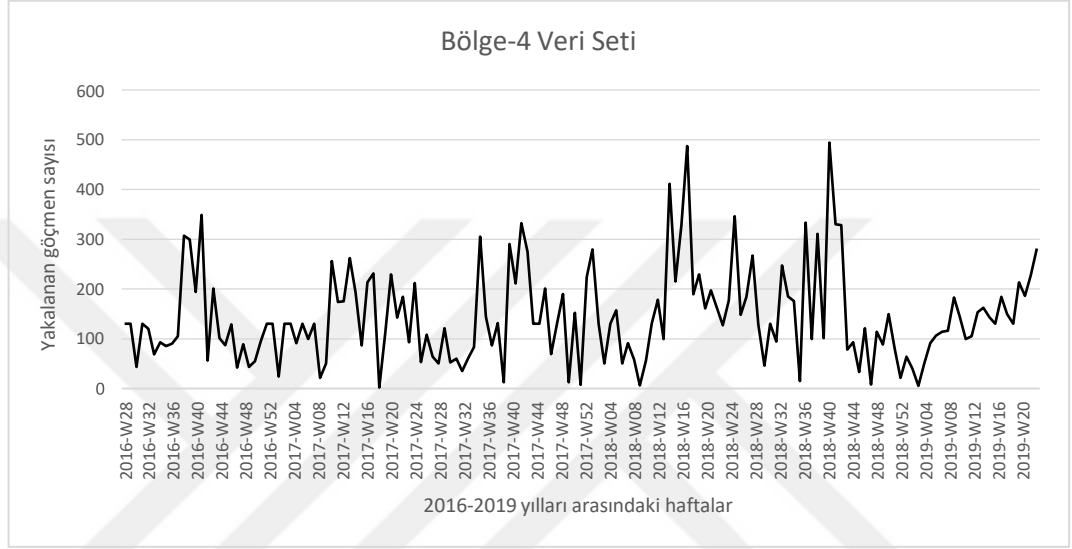
Tüm fonksiyonları oluşturmak için, küme sayısı 2'den 4'e ayarlandığında ve bulanık indeks parametresi 0,5'lik artış oranıyla 1,5'ten 2,5'e ayarlandığında $Z_{doğrulama}$ matrisi kullanılarak FCM gerçekleştirilir. FCA-MFFs algoritması, Bölge-4 veri kümesi için bu koşullar altında en iyi işlevi arar. Tüm iterasyonlar arasında en düşük RMSE değerine sahip fonksiyon, küme sayısı 4 ve bulanık indeks parametresi 2,5 olduğunda elde edilir. Demek ki metotlar 4 küme altında toplanmış yani 4 fonksiyon elde etmiş oluyoruz. Bu fonksiyon sırasıyla MFF_1 , MFF_2 , MFF_3 ve MFF_4 'tür ve bunlardan biri en düşük RMSE değerine sahip MFF_{best} 'tir. Yöntemlerin fonksiyonlardaki ağırlıkları ve $Z_{doğrulama}$ için fonksiyonların RMSE değerleri Çizelge 6.20'de verilir.

Çizelge 6.20, en iyi MFF'nin (MFF_{best}) RMSE açısından ilk fonksiyon olduğunu ortaya koymaktadır. Bu nedenle, tahminler Z_{test} için MFF_1 kullanılarak belirlenir. Çizelge 6.21'de verilen Z_{test} 'i elde etmek için 155 veriden ilk 129'u kullanılarak mevcut yöntemler eğitilir ve son 26 gözlem olan test veri seti (X_{test}) tahmin edilir. Z_{test} matrisi ve MFF_1 fonksiyonu kullanılarak MFF_{best} , Denklem (6.15)'deki gibi elde edilir. ve FCA-MFFs ve diğer mevcut yöntemler için tahmin sonuçları Çizelge 6.22'de verilir.

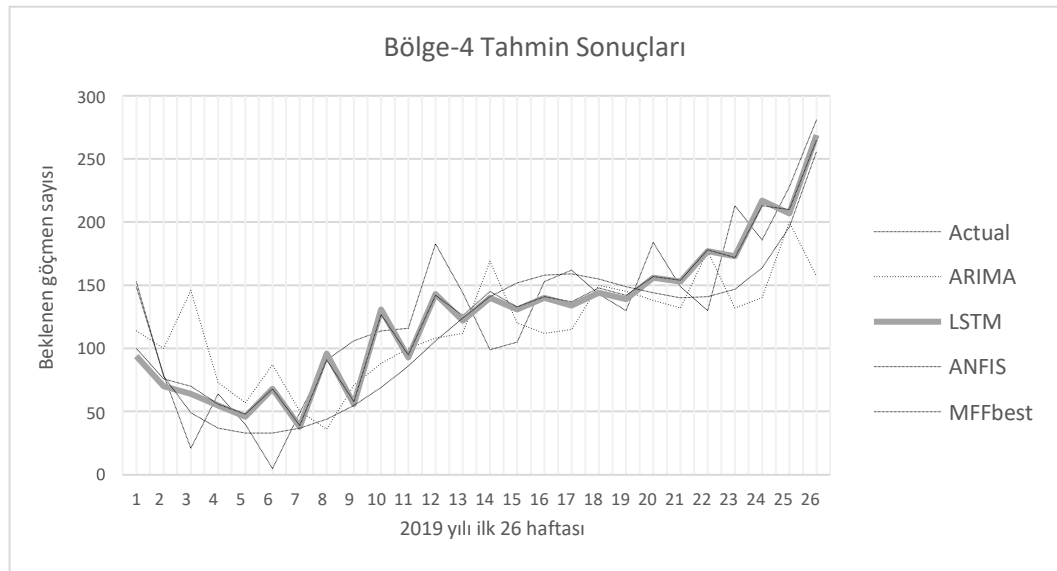
$$MFF_{best} = MFF_1(Z_{test}) = 0.001 \times ARIMA + 0.999 \times LSTM \quad (6.15)$$

Bölge-4 veri kümesi için, LSTM yöntemi, Denklem (6.15)'te görüldüğü gibi 0,999 ağırlıkla önerilen FCA-MFFs yöntemine en büyük katkıyı sağlar. LSTM yöntemini 0,001 ağırlıkla ARIMA takip etmektedir. Bu veri seti için SES, HES, ANN, ANFIS ve R-T1FF önerilen yöntemlere katkı sağlamamaktadır.

RMSE değerleri açısından en iyi tahmin sonuçlarının önerilen yöntemden elde edildiği açıktır. Yöntemlerin tahmin sonuçları Şekil 6.18'de gösterilmektedir.



Şekil 6.17 : Bölge-4 verileri.



Şekil 6.18 : Tahmin sonuçlarının karşılaştırılması.

Çizelge 6.19 : Bölge-4 için $Z_{doğrulama}$ veri seti.

Gözlem No	$X_{doğrulama}$	ARIMA	SES	HES	ANN	ANFIS	LSTM	R-TIFFs
104	105	183	108	114	177	135	202	194
105	148	178	110	200	200	121	184	194
106	174	174	109	215	216	123	162	194
107	271	176	117	202	218	136	273	194
108	205	179	129	172	250	156	145	194
109	92	188	158	354	116	181	215	194
110	174	188	168	179	177	209	231	194
111	198	178	152	217	198	235	205	194
112	227	180	157	272	315	256	208	194
113	228	183	165	231	141	268	176	194
114	405	187	178	207	230	270	191	194
115	313	189	188	141	131	264	282	194
116	437	204	233	202	218	251	202	194
117	293	205	249	182	250	235	227	194
118	227	215	288	201	86	217	164	194
119	174	209	289	131	204	199	195	194
120	86	200	276	116	217	182	104	194
121	275	191	255	145	262	168	165	194
122	90	180	220	182	144	157	81	194
123	68	189	232	256	223	147	158	194
124	91	179	203	296	160	141	72	194
125	169	172	175	116	192	136	196	194
126	288	170	158	138	215	133	254	194
127	57	175	160	121	225	131	121	194
128	204	188	186	139	167	131	203	194
129	206	176	160	123	231	132	264	194

Çizelge 6.20 : Yöntemlerin fonksiyonlardaki ağırlıkları ve $Z_{doğrulama}$ için RMSE değerleri.

Yöntem	MFF_1	MFF_2	MFF_3	MFF_4
ARIMA	0,001	0,301	0,018	0,004
SES	0,000	0,000	0,727	0,000
HES	0,000	0,000	0,000	0,857
ANN	0,000	0,249	0,048	0,037
ANFIS	0,000	0,140	0,203	0,099
LSTM	0,999	0,000	0,000	0,000
R-TIFFs	0,000	0,309	0,004	0,003
RMSE	82,55*	90,17	97,79	111,41

Çizelge 6.21 : Bölge-4 için Z_{test} veri seti ve yöntemlerin RMSE değerleri.

X_{test}	ARIMA	SES	HES	ANN	ANFIS	LSTM	R-TIFFs
181	171	117	173	135	207	145	254
174	156	137	125	129	190	100	50
132	144	146	42	215	166	124	176
247	160	152	43	202	143	244	168
39	177	148	85	163	125	146	176
148	147	170	341	164	114	168	163
168	184	141	132	205	110	140	125
105	224	143	43	192	111	163	176
67	164	148	154	154	117	81	126
366	188	139	175	180	128	117	141
130	226	123	34	192	141	230	187
139	120	176	32	188	155	169	140
127	102	166	149	175	168	146	183
174	202	160	85	227	175	177	237
360	188	153	37	195	175	339	126
41	175	158	180	151	168	104	152
281	86	202	201	204	154	265	169
349	145	167	91	215	136	216	210
29	178	192	37	189	118	40	252
80	46	226	32	260	103	70	132
112	160	183	152	235	91	122	54
214	260	161	182	180	86	289	304
64	209	150	169	231	89	75	79
71	161	164	41	160	105	102	80
420	185	142	195	229	147	364	87
496	242	127	130	180	254	332	225
RMSE	123.65	134.68	147.79	124.16	117.35	77.39	133.98

Çizelge 6.22 : Bölge-4 veri seti için mevcut yöntemler ve FCA-MFFs tahmin sonuçları ve RMSE karşılaştırması.

Gözlem	ARIMA	SES	HES	ANN	ANFIS	LSTM	R-TIFFs	FCA-MFFs
181	171	117	173	135	207	145	254	145
174	156	137	125	129	190	100	50	101
132	144	146	42	215	166	124	176	125
247	160	152	43	202	143	244	168	244
39	177	148	85	163	125	146	176	147
148	147	170	341	164	114	168	163	169
168	184	141	132	205	110	140	125	141
105	224	143	43	192	111	163	176	164
67	164	148	154	154	117	81	126	81
366	188	139	175	180	128	117	141	118
130	226	123	34	192	141	230	187	230
139	120	176	32	188	155	169	140	170
127	102	166	149	175	168	146	183	146
174	202	160	85	227	175	177	237	178
360	188	153	37	195	175	339	126	339
41	175	158	180	151	168	104	152	105
281	86	202	201	204	154	265	169	265
349	145	167	91	215	136	216	210	217
29	178	192	37	189	118	40	252	41
80	46	226	32	260	103	70	132	71
112	160	183	152	235	91	122	54	123
214	260	161	182	180	86	289	304	290
64	209	150	169	231	89	75	79	76
71	161	164	41	160	105	102	80	103
420	185	142	195	229	147	364	87	365
496	242	127	130	180	254	332	225	333
RMSE	123.65	134.68	147.79	124.16	117.35	77.39	133.98	77.34*

6.2.1.6 Genel değerlendirme ve yorum

Her bir veri seti için uygulanan yöntemlerden elde edilen RMSE performansları özet olarak Çizelge 6.23'de sunulur. Çizelge 6.23'de görüldüğü gibi, önerilen FCA-MFFs yöntemi tüm veri setlerinde en düşük RMSE değerine sahiptir ve daha doğru tahmin sonuçları verir. Aynı şekilde uygulamada ele alınan tüm veri setlerinde LSTM yöntemi FCA-MFFs'den sonra en iyi tahmin sonuçlarını vermektedir. Buna göre LSTM, tüm veri setlerinde FCA-MFFs yönteminde en büyük katkıyı sağlar. Ayrıca, FCA-MFFs için çalışma süresi ile hesaplama karmaşıklığı Çizelge 6.24'de sunulmaktadır.

Bu çalışma ile Türkiye'de Sahil Güvenlik Komutanlığı'nın göçmen tespitini daha etkin hale getirmek amacıyla yürütülen bir milli güvenlik projesi kapsamında, deniz hattındaki göçmen sayısı ilk kez tahmin ediliyor. Böylece göç hareketi gerçekleşmeden haftalık olarak tahmin edilerek, göçe müdahalenin ulusal güvenlik açısından işlerliğinin sağlanması amaçlanmaktadır.

Çalışmada öncelikle gerçek dünya tarihsel gözlem verilerinden oluşan beş farklı veri seti üzerinde geleneksel ve farklı anlayışlara sahip alternatif olan mevcut yedi tahmin yöntemi uygulanır ve COVID-19 nedeniyle 2020 ve 2021 verileri henüz elde edilemediği için haftalık olarak 2019'un ilk yarısı için göçmen sayısı tahminleri elde edilir. Sonuçlar, her tahmin yönteminin farklı performansa sahip olduğunu göstermektedir. Bu, her veri kümesi için tahmin yöntemlerini birleştirerek daha doğru ve güvenilir sonuçlar elde etmeyi mümkün kılar. Bu nedenle, daha doğru sonuçlar elde etmek ve çalışmanın başarısını artırmak için, mevcut tahmin yöntemlerini birleştiren FCM tabanlı bir yaklaşım olan önerilen yöntem FCA-MFFs uygulanmaktadır.

Sonuçlar, FCA-MFFs yönteminin diğer mevcut yöntemlerin performansını iyileştirebileceğini göstermektedir. Beş veri kümesinin tümü için RMSE değerleri açısından tahmin için en iyi performanslar FCA-MFF'lerden elde edilir. Ayrıca bu veri setleri için LSTM'nin mevcut diğer yöntemlere göre daha fazla performans sonucu verdiği ve önerilen FCA-MFFs yöntemine en fazla katkıyı sağladığı görülmektedir. Ancak, farklı veri kümeleri göz önüne alındığında diğer yöntemler daha iyi performans gösterebilir.

Çalışmada dikkat çeken bir diğer nokta ise bazı yöntemlerin uygulandıkları bölgelerde farklı performans göstermeleridir. Bunun nedeni bölgelere ait göçmen verilerinin

yapısının farklı olmasıdır. Aynı yöntem, farklı veri kümelerinde farklı performans gösterebilir. Bu nedenle, başlangıçta veri kümelerinin yapısı bilinmeyeceğinden, pratikte en uygun tahmin yöntemini seçmek zordur. Bu zorluğun üstesinden gelmek için, herhangi bir veri yapısıyla başa çıkabilen çeşitli tahmin yöntemlerini birleştirmek gerekir. Bu, çalışmada FCA-MFFs algoritması ile sağlanır.

Çalışmanın amacı, bir ulusal güvenlik projesi için haftalık olarak göçmen sayısını tahmin etmek olsa da, çalışma, ARIMA, SES, HES, ANFIS, ANN, LSTM ve R-T1FFs olmak üzere yedi farklı tahmin yönteminin ilk kez gerçek göçmen verileri üzerinde uygulanması bakımından benzersizdir. Ayrıca bu yedi yöntemin bir kombinasyonu olarak önerilen FCA-MFFs yöntemi de ilk kez uygulanmakta ve bu yöntemle daha güvenilir sonuçlar elde edilmektedir.

Bu aşama ile haftalık olarak elde edilen tahmin sonuçları kullanılarak verimli bir Karma Devriye Botu Rotalaması yapılması hedeflenir. Beklenen göçmen sayısının yoğunluğuna göre düzenlenen etkin bir devriye botu güzergahı, daha fazla göçmenin yakalanmasını ve dolayısıyla daha etkin devriyelerin yapılmasını sağlar.

Çizelge 6.23 : Bölgeler ve tahmin yöntemlerinin RMSE değerleri özet tablosu.

Bölgeler ve Tahmin Yöntemlerinin RMSE Değerleri					
Yöntem	Kıta Sahanelığı	Bölge-1	Bölge-2	Bölge-3	Bölge-4
ARIMA	220,02	25,01	122,22	38,06	123,65
SES	210,01	23,75	122,07	40,63	134,68
HES	240,82	80,42	133,58	88,38	147,79
ANN	219,93	43,65	184,7	74,68	124,16
ANFIS	198,93	24,56	118,92	33,61	117,35
LSTM	160,20	22,72	83	30,47	77,39
R-T1FFs	182,69	23,29	131,59	42,18	133,98
FCA-MFFs	159,85*	21,16*	82,84*	30,26*	77,34*

Çizelge 6.24 : FCA-MFFs için hesaplama karmaşıklığı.

FCA-MFFS hesaplama süresi (sn)	
Kıta Sahanelığı	5,93
Bölge-1	5,11
Bölge-2	4,76
Bölge-3	5,16
Bölge-4	4,98

6.3 Devriye botu rotalaması

Çalışmanın bu aşamasında, Türkiye'de ilk kez bir deniz hattı güvenlik projesi kapsamında devriye botu rotalama problemini inceliyoruz

2016-2019 yılları arasında göçmenlerin tespit edildiğı koordinatları K-Means Clustering algoritması ile kümeliyor ve elde edilen küme merkezlerini araç rotalama probleminin sıcak noktaları olarak belirliyoruz. FCA-MFFs ile elde ettiğimiz beklenen göçmen sayısı verilerini sıcak noktaların önem ağırlıkları olarak modele dahil ediyoruz. Tespit gücü, zaman ve devriye botu kullanılabilirliğinin sınırlamalarını göz önünde bulunduruyoruz ve sorunu çözmek için bir Meta Bulanık Fonksiyon Entegreli Karma Araç Rotalama (MFF-I-MVR) yaklaşımı öneriyoruz.

Çalışmamızı talep noktalarının belirlenmesi, önceliklendirilmesi ve rotalama olmak üzere üç ana aşamada ele alıyoruz.

Talep noktalarının belirlenmesi aşamasında, pilot bölge ile çalışarak, öncelikle devriye botu ile ziyaret edilmesi kritik kabul edilen talep noktaları olarak sıcak noktaları belirliyoruz. COVID-19 nedeniyle 2020 ve 2021 yılına ait verilere henüz ulaşamadığından, 2016-2019 yılları arasında tespit edilen göçmenlerin koordinatlarının K-Means algoritması ile kümelmesiyle elde edilen küme merkezlerini, yönlendirme probleminin sıcak noktaları olarak kabul ediyoruz.

Talep noktalarının önceliklendirilmesi aşamasında Bölüm 6.2'de elde edilen tahmin sonuçlarını kullanıyoruz.

Rotalama aşamasında iki senaryo ele alıyoruz. İlk olarak, pilot alan için geleneksel KARP modelini çözüyoruz. Bu yaklaşımla tüm sıcak noktaları ziyaret edilir ve amaç,

yönlendirmede toplam Öklid mesafesini en aza indirmektir. Daha sonra, problemi kaynakların sınırlı olduğu kısmi bir rota modeli olarak ele alıyoruz. Farklı önem ağırlık seviyeleri kullanarak, ağırlıklı Tespit Gücü ve kat edilen mesafe olmak üzere iki kriter ile verimli çözümler üretiyoruz. Bunu başarmak için, ziyaret edilen etkin noktaların önem ağırlıklarının toplamını “Tespit Gücü” kısıtı olarak modele dahil ediyoruz. Kıt kaynak durumunda, kullanılan devriye botu sayısı üzerindeki kısıtlamaların varlığında ağırlıklı tespit gücünü en üst düzeye çıkaran bir yaklaşım öneriyoruz ve bu modeli çözüyoruz. Böylece, beklenen göçmen sayısına göre maksimum tespit sağlayan haftalık bir devriye rotası elde ediyoruz. Ayrıca, farklı tespit gücü ve kat edilen mesafe politikaları altında faydalı olabilecek birden fazla verimli çözüm yaratıyoruz.

Uygulamamız, Türkiye’de deniz yolu üzerindeki göçmen sayısını tahmin etme ve buna bağlı devriye botu planlama alanında öncüdür ve devriye araçları için etkin noktaları ve verimli rotaları belirlemek için geçmiş göç verilerini kullandığından değerlidir. Ayrıca, sorunu farklı senaryolar altında ele alarak, farklı Tespit Gücü ve kaynak kullanım politikaları altında faydalı olabilecek birden fazla verimli çözüm sunan Çok Kriterli Karar Verme (ÇKKV) entegre perspektifi literatüre bir katkıdır.

6.3.1 Önerilen devriye botu rotalaması yaklaşımı (MFF-I-MVRP)

6.3.1.1 Talep noktalarının belirlenmesi

Geleneksel KARP’larda ziyaret edilecek talep noktaları bellidir. Bu aşamada öncelikle bir KARP’da talep noktalarının belirlenmesi sürecini ele alıyoruz. Bu talep noktalarını belirlemek için pilot bölgede yakalanan göçmenlerin tarihsel koordinat verilerini kümeliyoruz ve küme merkezlerini sıcak noktalar olarak kullanıyoruz. Uygulaması kolay olduğu ve büyük ölçekli verileri hızlı ve etkili bir şekilde kümeleyebildiği için bu aşamada K-Means algoritmasını kullanıyoruz.

Bu çalışmanın pilot bölgesi Şekil 6.19’da verilmektedir ve Türkiye kıta sahanlığı içerisinde yer alan ve 4 ayrı askeri limanın hakim olduğu bir alandır.

Şekil 6.19 : Pilot bölge gösterimi.

Bu çalışma için Sahil Güvenlik Komutanlığı envanterinde bulunan 7 farklı devriye botu kullanıyoruz. Devriye botlarının menzilleri ve ait oldukları askeri liman bilgileri Çizelge 6.25'de sunulmuştur.

Çizelge 6.25 : Devriye botu verileri.

Devriye Botu No	Menzil (km)	Askeri Liman
1	8000	Port-1
2	100	Port-2
3	200	Port-2
4	400	Port-3
5	300	Port-3
6	400	Port-4
7	8000	Port-4

2016-2019 yılları arasında göçmenlerin tespit edildiği ve Şekil 6.19'da gösterilen pilot bölgeyi ele alıyoruz. Mevcut koordinatların tamamını ziyaret etmek mümkün olmayacağı için bu koordinatların K-Means algoritması yardımıyla kümelenmesi ve

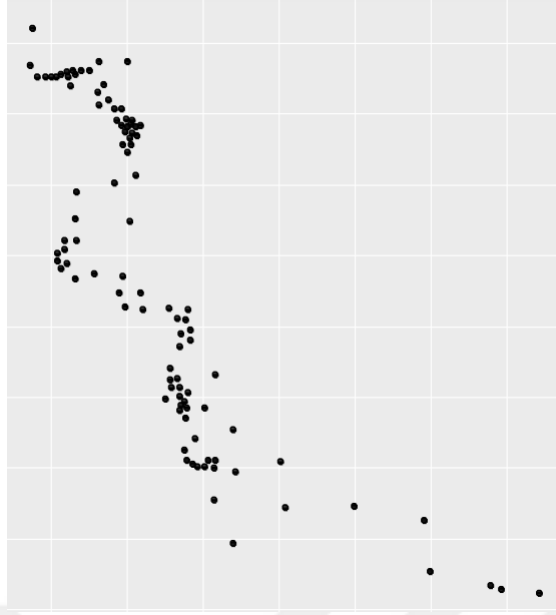
küme merkezlerinin sıcak noktalar olarak nitelendirilerek yönlendirme yapılmasını öneriyoruz.

Şekil 6.19'daki tüm koordinatları sırasıyla küme sayılarını 20, 30, 40, 50, 80, 100, 150, olarak kümeliyoruz ve küme merkezleri elde ediyoruz. Her kümeleme için küme merkezleri ve ilgili koordinatın merkezden uzaklığını hesaplıyoruz. Alternatif kümelemelerde, küme içi noktaların küme merkezinden ortalama uzaklığını (d_{AV}), küme içi noktaların küme merkezine olan maksimum mesafesini (d_{MAX}), küme içi noktaların küme merkezine olan minimum mesafesini (d_{MIN}) ve küme içi noktaların küme merkezinden uzaklığının Standart Sapması (d_{STD})'nı da inceleriz. Her bir kümeleme alternatifi için elde edilen sonuçlar Çizelge 6.26'da verilmiştir.

Çizelge 6.26 : Kümeleme alternatifleri.

Küme no	d_{AV} (km)	d_{MAX} (km)	d_{MIN} (km)	d_{STD} (km)
k=20	7,12	59,12	0	6,02
k=30	5,04	55,02	0	5,33
k=40	5,01	47,41	0	3,21
k=50	4,97	47,13	0	3,18
k=80	4,46	46,55	0	3,06
k=100*	3,01*	46,03	0	2,49
k=150	2,87	33,79	0	2,21
k=200	2,33	32,89	0	2,02

Küme içi noktaların küme merkezine olan ortalama mesafesinin 3 km olması alanında uzman personel (karar verici) tarafından yeterli görülmektedir. Uzman görüşlerine göre bu mesafe denizde tekrar bir göçmenle karşılaşmak açısından uygun bir mesafedir. Bu nedenle, küme sayısı K=100 optimal alıyoruz. Ayrıca, K=100 ile standart sapma 2,49 km'dir, bu yeterince düşük bir değerdir. 100 küme merkezi Şekil 6.20'de görsel olarak verilmektedir. Şekil 6.20'de gösterilen bu 100 noktayı problemimizin sıcak noktaları olarak alıyoruz.



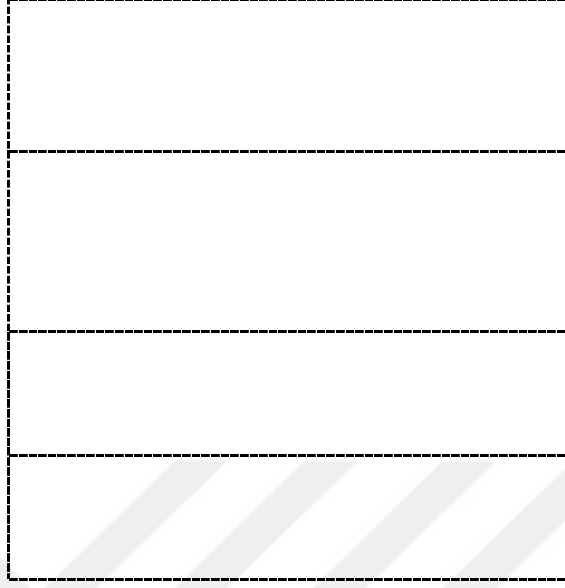
Şekil 6.20 : Küme merkezleri (sıcak noktalar).

K=100 için elde edilen kümeler ve merkezleri detaylı olarak Çizelge 6.27’de verilmektedir. Proje için her noktanın tanımı, adresi ve koordinatları GPS formatında sağlanır. Ancak pilot bölge ve genel olarak proje ile ilgili bazı veriler güvenlik nedeniyle bu yazıda paylaşılmamaktadır.

Çözüm aşamasında pilot bölge 4 ayrı askeri limanın kontrolünde olduğundan bu limanların Karar Verici (KV)’lerinin farklı olduğunu düşünüyoruz. Bu nedenle KV’lerin kendi sorumluluk alanlarında karar verebilmeleri için pilot bölgedeki 100 sıcak noktayı Şekil 6.2 ile uyumlu olacak şekilde 4 gruba ayırıyor ve her grubu kendi içinde çözüyoruz.

Gruplama yapılırken devriye araçlarının güzergah yönü, askeri limanların sorumluluk alanları, göç olayının doğası ve geçmiş göç verilerinin yoğunluğu göz önünde bulundurulur. Türkiye’de denizyolunda iki ana yasadışı göç türü vardır. Birincisi kaçakçılık, ikincisi ise Doğu ülkelerindeki savaşlardan kaçıp Batı ülkelerine geçmeden önce Türkiye’ye sığınmak. Her iki durumda da göçün Türkiye’nin batı kıyısından Yunanistan’a doğru olması kuvvetle muhtemeldir. Bu nedenle, Türkiye’nin jeopolitik konumu nedeniyle göçmenlerin kaçış yollarının enlem yönünde olacağını öngörerek, koordinatların enlemsel olarak gruplandırılmasını öneriyoruz. Tüm bu hususlar göz

önüne alındığında, farklı limanlar tarafından devriye gezilecek olan 4 gruba ait gösterim Şekil 6.21’de, detaylı bilgiler Çizelge 6.28’de gösterilmektedir.



Şekil 6.21 : Sıcak noktaların gruplandırılması.

Çizelge 6.27 : Sıcak noktaların gruplandırılmasına ait bilgiler.

Sıcak Nokta No	Liman
1, 20, 24, 27, 46, 58, 88, 89, 90	Liman -1
10, 15, 18, 19, 34, 35, 41, 45, 48, 55, 56, 61, 63, 64, 65, 66, 67, 77, 80, 84, 85, 87, 91, 96, 100	Liman -2
2, 7, 8, 11, 16, 17, 23, 29, 31, 32, 33, 37, 39, 42, 43, 44, 50, 52, 54, 68, 69, 71, 73, 76, 78, 79, 82, 83, 94, 95, 97, 98, 99	Liman -3
3, 4, 5, 6, 9, 12, 13, 14, 21, 22, 25, 26, 28, 30, 36, 38, 40, 47, 49, 51, 53, 57, 59, 60, 62, 70, 72, 74, 75, 81, 86, 92, 93	Liman -4

Çizelge 6.28 : Küme merkezleri.

Küme No	Boylam (Derece)	Enlem (Derece)	Küme No	Boylam (Derece)	Enlem (Derece)
küme 1	27,95	36,67	küme 51	26,00	39,51
küme 2	26,66	38,80	küme 52	26,27	38,34
küme 3	26,73	39,08	küme 53	26,52	39,22
küme 4	26,75	39,11	küme 54	26,26	38,38
küme 5	26,57	39,27	küme 55	27,04	37,39
küme 6	26,28	39,46	küme 56	27,16	37,88
küme 7	26,23	38,36	küme 57	26,16	39,44
küme 8	26,20	38,24	küme 58	27,34	36,94
küme 9	26,44	39,45	küme 59	26,37	39,46
küme 10	27,10	37,91	küme 60	26,84	39,12
küme 11	26,32	38,39	küme 61	27,11	37,46
küme 12	26,24	39,44	küme 62	26,64	39,35
küme 13	26,29	39,43	küme 63	27,23	37,10
küme 14	26,01	39,74	küme 64	27,00	37,97
küme 15	27,17	37,96	küme 65	27,06	37,53
küme 16	26,21	38,28	küme 66	27,18	37,73
küme 17	26,56	38,71	küme 67	27,12	37,71
küme 18	27,17	37,28	küme 68	26,24	38,26
küme 19	27,23	37,77	küme 69	26,23	38,31
küme 20	27,18	37,01	küme 70	26,81	39,09
küme 21	26,20	39,44	küme 71	26,77	39,00
küme 22	26,05	39,43	küme 72	26,72	39,03
küme 23	26,34	38,47	küme 73	26,31	38,60
küme 24	27,69	36,94	küme 74	26,61	39,69
küme 25	26,70	39,22	küme 75	26,46	39,49
küme 26	26,33	39,46	küme 76	26,24	38,19
küme 27	27,26	36,95	küme 77	27,13	37,32
küme 28	26,77	39,05	küme 78	26,78	37,98
küme 29	26,70	38,99	küme 79	26,76	38,51
küme 30	26,78	39,14	küme 80	27,12	37,25

Küme No	Boylam (Derece)	Enlem (Derece)	Küme No	Boylam (Derece)	Enlem (Derece)
küme 31	26,73	38,75	küme 81	26,60	39,51
küme 32	26,35	38,53	küme 82	26,73	38,94
küme 33	26,38	38,75	küme 83	26,74	38,12
küme 34	27,17	37,34	küme 84	27,15	37,79
küme 35	27,20	37,33	küme 85	27,23	37,92
küme 36	26,64	39,23	küme 86	26,12	39,43
küme 37	26,25	38,42	küme 87	27,13	37,52
küme 38	26,68	39,15	küme 88	27,42	37,53
küme 39	26,85	38,84	küme 89	28,87	36,59
küme 40	26,71	39,11	küme 90	29,58	36,18
küme 41	27,22	37,82	küme 91	27,19	37,38
küme 42	26,20	38,31	küme 92	26,55	39,38
küme 43	26,42	38,37	küme 93	26,90	39,55
küme 44	26,73	38,99	küme 94	26,78	38,95
küme 45	27,15	37,36	küme 95	26,23	38,24
küme 46	27,39	36,99	küme 96	27,18	37,44
küme 47	26,51	39,33	küme 97	26,60	38,16
küme 48	27,14	37,40	küme 98	26,36	38,12
küme 49	26,74	39,16	küme 99	26,28	38,24
küme 50	26,35	38,20	küme 100	27,39	37,30

6.3.1.2 Senaryo 1 : Geleneksel karma araç rotalama modeli

Bu senaryoda, kaynakların pilot alandaki tüm sıcak noktaları ziyaret etmek için yeterli olduğunu varsayıyoruz. 4 askeri limanın her biri için KARP modelini Bölüm 4'te yer alan Denklem (4.1)-(4.7) ile çözüyoruz ve buna göre minimum mesafe çözümlerini belirliyoruz.

Daha önce belirtildiği gibi, Öklid mesafesini kullanarak sıcak nokta çiftleri arasındaki mesafeleri hesaplıyoruz. Türkiye'nin coğrafi konumuna göre iki boylam arası 86 km, iki enlem arası 111 km olarak alınmıştır. Çizelge 6.29, her bir bölgede hangi devriye botlarının yönlendirildiğini ve kat ettikleri mesafeleri göstermektedir.

Her bir askeri liman alanındaki rotalar, Çizelge 6.30'da sunulmaktadır. Özetle, Liman-1 tarafından 8000 km menzilli bir devriye botu, Liman-2 tarafından 100 ve 200 km'lik botlar, Liman-3 tarafından 200 km ve 300 km ve Liman-4 tarafından ise 1000 km'lik bot kullanılmaktadır. Liman-4'te 7 numaralı bot menzili yeterli olmadığı için kullanılmamaktadır. Limanlara ait çözümlerde kat edilen toplam mesafeler sırasıyla 538,15 km, 259,77 km, 360,76 km ve 580,69 km'dir.

Çizelge 6.29 : Senaryo 1 atamaları.

Askeri Liman No	Atanan Devriye Botu	Uzaklık
1	1	538,15 km
2	2, 3	259,77 km
3	4, 5	360,76 km
4	6	580,69 km

Ardından, ağırlıklı tespit gücünü içeren sonuçlarımızı sunacağız. Ancak bu modeli formüle etmeden önce ziyaret edilecek sıcak noktaların önem ağırlıklarını belirlememiz gerekir. Bir sonraki bölümde, bu amaç için FCA-MFFs algoritmasını kullanıyoruz.

Çizelge 6.30 : Senaryo 1 çözümleri.

LİMAN-1			LİMAN-2			LİMAN-3			LİMAN-4		
<i>i</i> 'den	<i>j</i> 'ye	Atanan devriye botu	<i>i</i> 'den	<i>j</i> 'ye	Atanan devriye botu	<i>i</i> 'den	<i>j</i> 'ye	Atanan devriye botu	<i>i</i> 'den	<i>j</i> 'ye	Atanan devriye botu
Port-1	90	1	Port-2	100	2	Port-3	78	4	Port-4	70	6
90	1	1	100	35	2	78	Port-3	4	70	28	6
1	24	1	35	34	2	Port-3	97	5	28	72	6
24	46	1	34	91	2	97	98	5	72	3	6
46	58	1	91	96	2	98	50	5	3	4	6
58	27	1	96	87	2	50	76	5	4	40	6
27	20	1	87	66	2	76	99	5	40	38	6
20	88	1	66	19	2	99	68	5	38	53	6
88	89	1	19	41	2	68	95	5	53	5	6
89	Port-1	1	41	85	2	95	8	5	5	47	6
			85	15	2	8	16	5	47	92	6
			15	64	2	16	42	5	92	81	6
			64	10	2	42	69	5	81	75	6
			10	56	2	69	52	5	75	9	6
			56	84	2	52	7	5	9	59	6
			84	67	2	7	54	5	59	26	6
			67	65	2	54	37	5	26	13	6
			65	61	2	37	11	5	13	6	6
			61	55	2	11	43	5	6	12	6
			55	48	2	43	23	5	12	21	6
			48	45	2	23	32	5	21	57	6
			45	77	2	32	73	5	57	86	6
			77	18	2	73	33	5	86	22	6
			18	80	2	33	17	5	22	51	6
			80	Port-2	2	17	2	5	51	14	6
			Port-2	63	3	2	82	5	14	74	6
			63	Port-2	3	82	29	5	74	93	6
						29	44	5	93	62	6
						44	71	5	62	36	6
						71	94	5	36	25	6
						94	39	5	25	49	6
						39	31	5	49	30	6
						31	79	5	30	60	6
						79	83	5	60	Port-4	6
						83	Port-3	5			

6.3.1.3 Sıcak noktaların önem ağırlıklarının belirlenmesi

Bu aşamada, Bölüm 6.2.1’de FCA-MFFs ile elde edilen göçmen sayısı tahminlerini kullanarak 2019 yılı 1.haftası için karma araç rotalama modelimizin talep noktaları olan sıcak noktaların önem ağırlıklarını hesaplıyoruz. KV’nin diğer haftalar için yeniden önem ağırlığı elde etmesine imkan sağlayan hesaplama adımlarını paylaşıyoruz.

Sıcak noktaların önem ağırlıklarını, ait oldukları bölgedeki haftalık tahmin sonucunun merkezi oldukları kümelerde yer alan küme içi noktaların geçmiş gözlem verileri ile orantılı olarak dağıtılmasıyla elde etmeyi anlamlı buluyoruz. Bunu yapmak için sırasıyla aşağıdaki adımları izliyoruz:

Adım 1. Sıcak noktaları Çizelge 6.27’da yapılan bölge gruplandırması ile paralel olacak şekilde gruplandırıyoruz. Bu gruplandırma tekrar Çizelge 6.31’de verilir.

Adım 2. Bölüm 6.2’de elde edilen haftalık bazda göçmen sayısı tahminlerini alıyoruz. Bu tahminlerden 2019 yılı 1.haftası için elde edilen sonuçları tutuyoruz. Bu sonuçlar Çizelge 6.32’de verilir.

Adım 3. Her sıcak nokta için, merkezi oldukları kümede yer alan noktalarda yakalanan geçmiş göçmen sayısı verilerini elde ediyoruz. Bu veriler Çizelge 6.33’de verilir.

Adım 4. Çizelge 6.31’de yer alan beklenen göçmen sayısı verilerini, Çizelge 6.32’de sunulan geçmiş gözlem verileri ile orantılı olarak dağıtıyoruz ve sıcak noktaların 2019 yılı 1.haftası için önem ağırlığını Çizelge 6.34’deki şekilde elde etmiş oluyoruz.

Adım 5. Beklenen göçmen sayısı haftalık olarak değişeceğinden, KV’nin Adım 1-4’ü her hafta tekrarlayarak sıcak noktaların önem ağırlıklarını güncellemesini öneriyoruz.

Çizelge 6.31 : Enlem-Bölge eşleşmesi.

Enlem	Bölge	Askeri Liman	Sıcak Nokta No
26°	Bölge-1	Liman-1	1, 20, 24, 27, 46, 58, 88, 89, 90
32°			
35°			
36°			
37°	Bölge-2	Liman-2	10, 15, 18, 19, 34, 35, 41, 45, 48, 55, 56, 61, 63, 64, 65, 66, 67, 77, 80, 84, 85, 87, 91, 96, 100
38°	Bölge-3	Liman-3	2, 7, 8, 11, 16, 17, 23, 29, 31, 32, 33, 37, 39, 43, 44, 50, 52, 54, 68, 69, 71, 73, 76, 78, 79, 82, 83, 94, 95, 97, 98, 99, 42
39°	Bölge-4	Liman-4	3, 4, 5, 6, 9, 12, 13, 14, 21, 22, 25, 26, 28, 30, 36, 38, 40, 47, 49, 51, 53, 57, 59, 60, 62, 70, 72, 74, 75, 81, 86, 92, 93
40°			
41°			
42°			
43°			

Çizelge 6.32 : Gruplar ve tahmin değerleri.

Bölge	Sıcak Nokta No	2019/1.haftası için beklenen göçmen sayısı
Bölge-1	1, 20, 24, 27, 46, 58, 88, 89, 90	42
Bölge-2	10, 15, 18, 19, 34, 35, 41, 45, 48, 55, 56, 61, 63, 64, 65, 66, 67, 77, 80, 84, 85, 87, 91, 96, 100	97
Bölge-3	2, 7, 8, 11, 16, 17, 23, 29, 31, 32, 33, 37, 39, 43, 44, 50, 52, 54, 68, 69, 71, 73, 76, 78, 79, 82, 83, 94, 95, 97, 98, 99, 42	100
Bölge-4	3, 4, 5, 6, 9, 12, 13, 14, 21, 22, 25, 26, 28, 30, 36, 38, 40, 47, 49, 51, 53, 57, 59, 60, 62, 70, 72, 74, 75, 81, 86, 92, 93	145

Çizelge 6.33 : Sıcak noktalar ve küme içi geçmiş gözlem verileri.

Sıcak Nokta No (<i>i</i>)	Küme içi gözlem değerleri (geçmiş göçmen sayısı)	Sıcak Nokta No (<i>i</i>)	Küme içi gözlem değerleri (geçmiş göçmen sayısı)	Sıcak Nokta No (<i>i</i>)	Küme içi gözlem değerleri (geçmiş göçmen sayısı)	Sıcak Nokta No (<i>i</i>)	Küme içi gözlem değerleri (geçmiş göçmen sayısı)
1	1965	26	8436	51	906	76	1320
2	962	27	4156	52	1694	77	1599
3	2447	28	2793	53	318	78	2419
4	2891	29	1041	54	2011	79	125
5	1537	30	2251	55	276	80	416
6	4201	31	1160	56	2856	81	775
7	1470	32	1920	57	1912	82	1218
8	2166	33	698	58	5757	83	2504
9	1714	34	3982	59	4063	84	620
10	1569	35	1952	60	660	85	1360
11	1066	36	662	61	1997	86	1161
12	3018	37	552	62	197	87	1490
13	4242	38	1291	63	1532	88	84
14	686	39	390	64	1604	89	2191
15	1242	40	2651	65	1179	90	3338
16	3210	41	853	66	513	91	1400
17	425	42	2394	67	512	92	818
18	1026	43	267	68	1749	93	158
19	857	44	1663	69	1959	94	1064
20	964	45	1320	70	1015	95	1236
21	1123	46	5708	71	2357	96	645
22	801	47	1417	72	2529	97	984
23	1383	48	877	73	1774	98	763
24	858	49	1960	74	178	99	2540
25	731	50	910	75	2788	100	653

Çizelge 6.34 : Sıcak noktalar ve önem ağırlıkları.

Sıcak Nokta No (i)	beklenen göçmen sayısı	Küme içi gözlem değerleri	önem ağırlığı (q _i)	Sıcak Nokta No (i)	beklenen göçmen sayısı	Küme içi gözlem değerleri	önem ağırlığı (q _i)
1	42	1965	3,30	51	145	906	2,11
2	100	962	2,05	52	100	1694	3,57
3	145	2447	5,70	53	145	318	0,74
4	145	2891	6,73	54	100	2011	4,24
5	145	1537	3,58	55	97	276	0,83
6	145	4201	9,77	56	97	2856	8,57
7	100	1470	3,10	57	145	1912	4,45
8	100	2166	4,57	58	42	5757	9,66
9	145	1714	3,99	59	145	4063	9,45
10	97	1569	4,68	60	145	660	1,54
11	100	1066	2,25	61	97	1997	5,99
12	145	3018	7,02	62	145	197	0,46
13	145	4242	9,87	63	97	1532	4,60
14	145	686	1,60	64	97	1604	4,81
15	97	1242	3,73	65	97	1179	3,54
16	100	3210	6,77	66	97	513	1,54
17	100	425	0,90	67	97	512	1,54
18	97	1026	3,08	68	100	1749	3,69
19	97	857	2,57	69	100	1959	4,13
20	42	964	1,62	70	145	1015	2,36
21	145	1123	2,61	71	100	2357	4,97
22	145	801	1,86	72	145	2529	5,88
23	100	1383	2,92	73	100	1774	3,74
24	42	858	1,44	74	145	178	0,41
25	145	731	1,70	75	145	2788	6,49
26	145	8436	19,62	76	100	1320	2,79
27	42	4156	6,98	77	97	1599	4,80
28	145	2793	6,50	78	100	2419	5,10
29	100	1041	2,20	79	100	125	0,26
30	145	2251	5,24	80	97	416	1,25
31	100	1160	2,45	81	145	775	1,80
32	100	1920	4,05	82	100	1218	2,57
33	100	698	1,47	83	100	2504	5,28
34	97	3982	11,95	84	97	620	1,86
35	97	1952	5,86	85	97	1360	4,08
36	145	662	1,54	86	145	1161	2,70
37	100	552	1,16	87	97	1490	4,47
38	145	1291	3,00	88	42	84	0,14
39	100	390	0,82	89	42	2191	3,68
40	145	2651	6,17	90	42	3338	5,60
41	97	853	2,56	91	97	1400	4,20
42	100	2394	5,05	92	145	818	1,90
43	100	267	0,56	93	145	158	0,37
44	100	1663	3,51	94	100	1064	2,25
45	97	1320	3,96	95	100	1236	2,61
46	42	5708	9,58	96	97	645	1,94
47	145	1417	3,30	97	100	984	2,08
48	97	877	2,63	98	100	763	1,61
49	145	1960	4,56	99	100	2540	5,36
50	100	910	1,92	100	97	653	1,96

6.3.1.4 Senaryo 2 : Tespit gücü sınırlarına sahip önem ağırlıklı rotalama modeli

Belirli bir rotadaki tüm sıcak noktaları ziyaret etmek mümkün olmadığında, aynı anda kat edilen toplam mesafeyi en aza indirmek ve toplam Tespit Gücü'nü en üst düzeye çıkarmak için biobjektif bir yaklaşım benimsiyoruz. Hedef olarak mesafeyi ve kısıtlama olarak Tespit Gücü'nü ele alarak Augmented ε -Constraint (Haimes vd. 1971) yöntemini kullanıyoruz. Tespit gücü seviyesi sınırlarını sistematik olarak değiştirerek, KV'lerin bilinçli bir karar vermesi için alternatif verimli çözümler elde ediyoruz.

Augmented ε -Constraint, iki veya daha fazla kriterle sahip problemler için ayrık çözümler üretme yöntemidir. Kriterlerden biri optimize edilirken diğerleri limitli kısıtlamalar olarak değerlendirilir. Bu limitler sistematik olarak değiştirilerek istenildiği kadar farklı çözüm üretilebilir. Matematiksel Denklem (6.16)-(6.18)'deki gibidir:

$$\text{Min } f_k(x) + c \sum_{i \neq k} f_i(x) \quad (6.16)$$

$$f_i(x) \leq \varepsilon_i, i \neq k \quad (6.17)$$

$$x \in X \quad (6.18)$$

Burada; $f_i(x)$ i . amaç fonksiyonunu temsil eder, ε_i , i amacını sınırlamak için kullanılan değeri temsil eder, c yeterince küçük bir pozitif sabittir ve X uygulanabilir noktalar kümesini temsil eder.

Bu bölümde, ilk olarak, Bölüm 4'te açıklanan geleneksel KARP modellerine hangi parametrelerin ve karar değişkenlerinin entegre edildiğini ve amaç fonksiyonu ve kısıtlamaların nasıl revize edildiğini açıklıyoruz. Ardından, önerdiğimiz yeni Meta Bulanık Fonksiyonlar Entegre Karma Araç Yönlendirme (MFF-I-MVR) yaklaşımının pratik faydalarını değerlendiriyoruz.

Bu bölümde öncelikle yaklaşımımız için geliştirilen KARP modelini aşağıdaki gibi tanıtalım:

Parametreler

N	Sıcak noktalar kümesi
K	Devriye araçları kümesi
B	Liman
d_{ij}	i ve j noktaları (sıcak noktalar ve liman) arasındaki Öklid mesafesi
f_k	k aracının menzili
q_i	Sıcak nokta i 'nin önem ağırlığı
Φ	Küçük bir pozitif katsayı
R_{LB}	Tespit gücü alt sınırı

Karar Değişkenleri

x_{ijk}	İkili değişken. Eğer i noktasından j noktasına k aracı ile gidiliyorsa “1” aksi halde “0” değerini alır.
y_i	Alt tur oluşumunu engelleyen değişken

$$\text{Min } \sum_{i \in NU\{B\}} \sum_{j \in NU\{B\}} \sum_{k \in K} d_{ij} x_{ijk} - \Phi \sum_{i \in NU\{B\}} \sum_{j \in NU\{B\}} \sum_{k \in K} q_i x_{ijk} \quad (6.19)$$

$$\sum_{i \in \{B\} \cup N \setminus \{j\}} x_{ijk} = \sum_{i \in \{B\} \cup N \setminus \{j\}} x_{jik} \quad j \in N \cup \{B\}, k \in K \quad (6.20)$$

$$\sum_{i \in NU\{B\}} \sum_{k \in K} x_{ijk} \leq 1 \quad j \in N \quad (6.21)$$

$$\sum_{i \in NU\{B\}} \sum_{j \in NU\{B\}} x_{ijk} d_{ij} \leq f_k \quad k \in K \quad (6.22)$$

$$\sum_{i \in NU\{B\}} \sum_{j \in NU\{B\}} \sum_{k \in K} q_i x_{ijk} \geq R_{LB} \quad (6.23)$$

$$y_i - y_j + (|N \cup \{B\}| - 1) x_{ijk} \leq |N \cup \{B\}| - 2 \quad i, j \in N \quad i \neq j, k \in K \quad (6.24)$$

$$1 \leq y_i \leq |N \cup \{B\}| - 1 \quad i \in N \quad (6.25)$$

$$x_{ijk} \in \{0,1\} \quad i, j \in N \cup \{B\}, k \in K \quad (6.26)$$

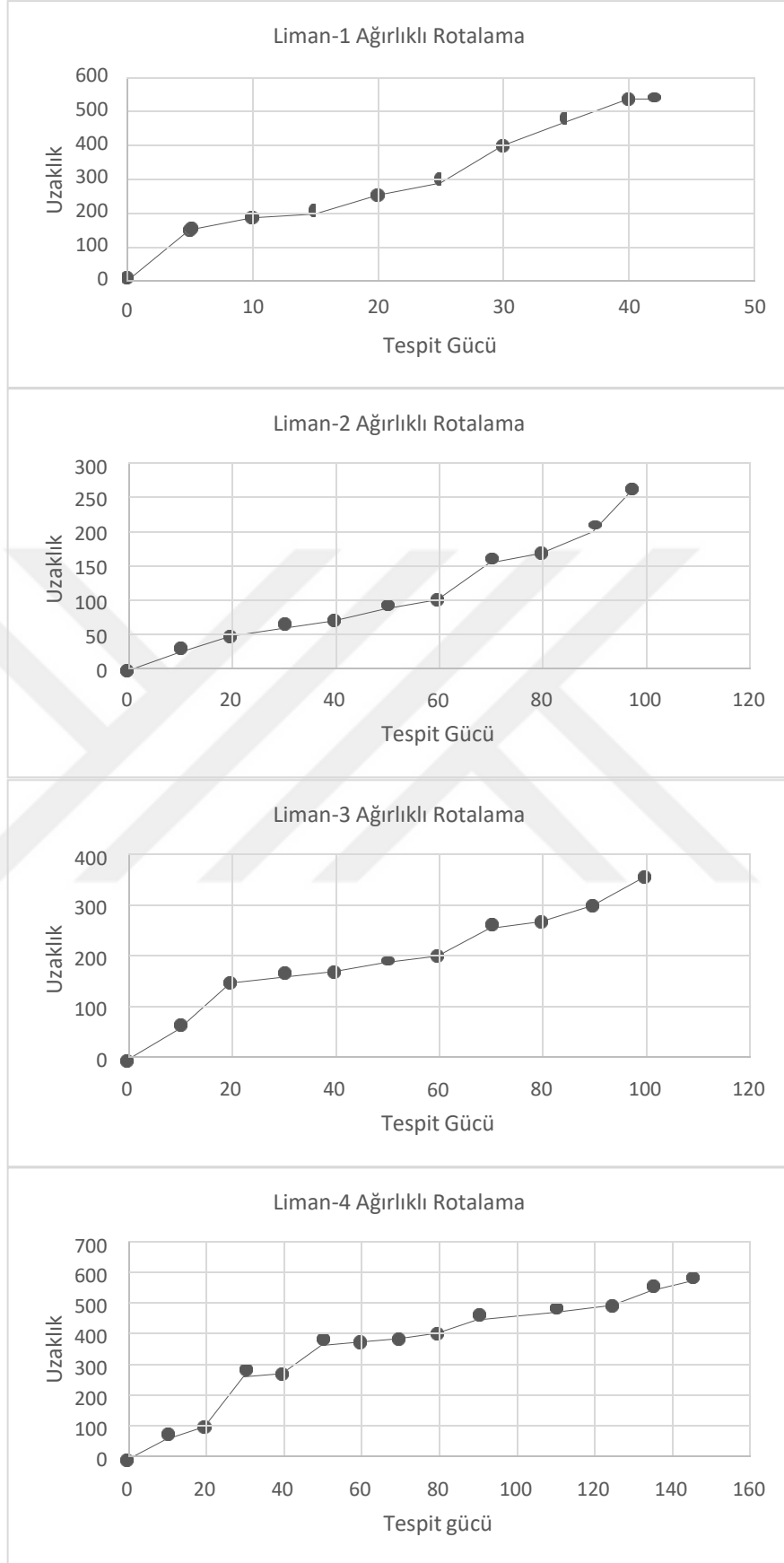


Amaç fonksiyonu (6.19), ziyaret edilen sıcak noktalar ve limanlar arasındaki Öklid mesafelerinin toplamını kısıtlamalara uyan en iyi seviyede en aza indirir. Kısıt (6.20), aynı devriye bptuyla bir noktaya giriş ve çıkış sağlar. Kısıt (6.21), her sıcak noktaya en fazla bir kere gidilmesini garanti eder. Kısıt (6.22) her devriye botunun kat ettiği toplam mesafenin botun menziline daha az olmasını sağlar. Kısıt (6.23) KV tarafından belirlenen tespit gücü gereksinimlerini karşılar. Tüm sıcak noktalara gidilmesi gerektiğinde, toplam tespit gücünün alt sınırı, soruna dahil olan tüm sıcak noktaların toplam önem ağırlığına ayarlanabilir. Kısıtlar (6.24)-(6.25) alt turları engeller. Kısıt (6.26), karar değişkenlerinin 0-1 tamsayı olmasını sağlar.

Model (6.19)-(6.26)'yı uygulamamıza özel parametrelerle çözüyoruz. Liman-1'de N , 9 sıcak nokta ve K , 1 devriye botu, Liman-2'de N , 25 sıcak nokta ve K , 2 devriye botu, liman-3'te N , 33 sıcak nokta ve K , 2 devriye botu, Liman-4'te N , 33 sıcak nokta ve K , 2 devriye botu içerir.

Devriye botlarının rotalarını tamamladıktan sonra kendi limanlarına dönmeleri gerekmektedir. Çizelge 6.33'te yer alan önem ağırlıkları q_i değerleri olarak kullanılmıştır. d_{ij} parametrelerinin değerleri, daha önce açıklandığı gibi hesaplanan uzaklık matrisi yardımıyla belirlenir. Φ sabiti 10^{-3} olarak alınır. Bu değer, mesafe yerine Tespit Gücü kriterinin daha küçük değerleri dikkate alınarak seçilir. Deneylerde parametrenin bu değerinin iyi çalıştığı gözlenmiştir.

Değişken Tespit Gücü ve mesafe değerleri ile üretilen verimli çözümler Şekil 6.22'de verilmektedir. Bu çözümlerin mesafe, tespit gücü ve atanan devriye botu bilgileri Çizelge 6.35'te detaylı olarak verilmektedir. Şekil 6.22, bu sonuçların bir özeti olarak etkin sınırı göstermektedir. Tüm bu çözümler, farklı öncelik ve koşullarda tercih edilebilecek mantıklı çözümlerdir. Ek olarak, Tespit Gücü ile kat edilen mesafe arasındaki dengeyi inceleyerek, KV, belirli bir Tespit Gücü seviyesini elde etmek için ne kadar ek mesafe kat edilmesi gerektiğini ve kaç ekstra devriye botu kullanılması gerektiğini gözlemleyebilir. Örneğin, KV, makul bir mesafe artışıyla tespit gücünde önemli bir gelişme sağlanabileceğine karar verirse, ilk plana ekstra devriye botları eklenebilir.



Şekil 6.22 : Etkin çözümler.

Çizelge 6.35 : Ağırlıklı rotalama çözümleri ve atamaları.

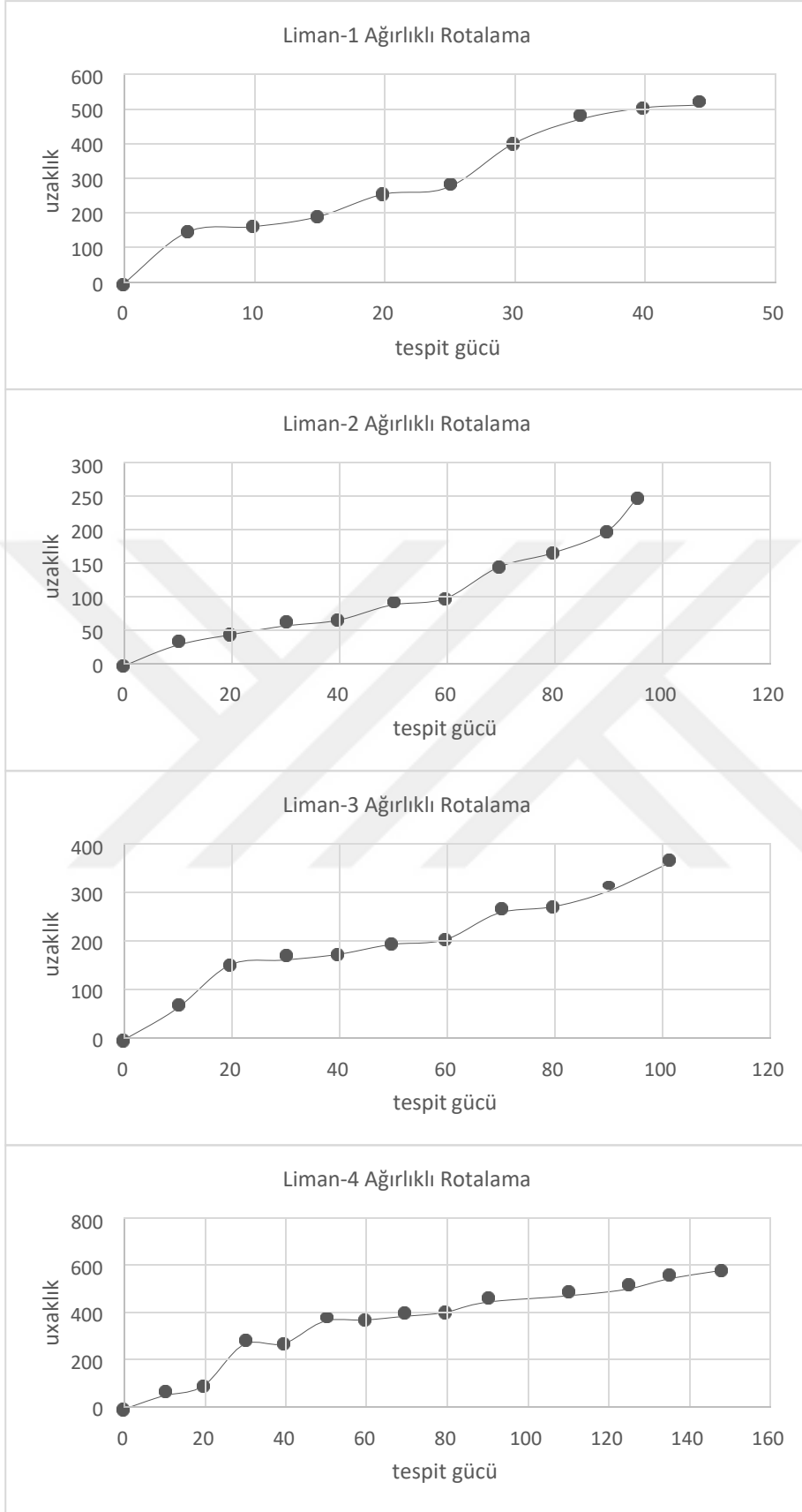
Limn	Tespit Gücü	Uzakhk (km)	Atanan araç(lar)
Limn-1	42	538,15	1
	40	534,35	1
	35	470,59	1
	30	400,78	1
	25	289,96	1
	20	254,70	1
	15	198,80	1
	10	187,12	1
	5	152,63	1
	0	0,00	-
Limn-2	97	259,77	2, 3
	90	203,50	2, 3
	80	171,90	3
	70	158,47	3
	60	104,64	3
	50	91,30	2 veya 3
	40	73,60	2 veya 3
	30	62,32	2 veya 3
	20	51,77	2 veya 3
	10	27,12	2 veya 3
0	0,00	-	
Limn-3	100	360,76	4, 5
	90	304,50	4, 5
	80	272,90	5
	70	259,11	5
	60	205,65	5
	50	192,30	4 veya 5
	40	174,60	4 veya 5
	30	163,00	4 veya 5
	20	152,25	4 veya 5
	10	61,06	4 veya 5
0	0,00	-	
Limn-4	145	580,69	6
	135	550,94	6
	125	503,97	6
	110	479,02	6
	90	455,14	6
	80	412,30	6
	70	394,60	6
	60	383,33	6
	50	372,00	6
	40	281,44	6 veya 7
	30	270,72	6 veya 7
	20	108,01	6 veya 7
	10	66,06	6 veya 7
	0	0,00	-

Ayrıca önem ağırlıklarındaki değişikliklerin etkilerini analiz etmek için duyarlılık analizi de yapıyoruz. Sonuçlarımız, FCA-MFFs ile hesaplanan ağırlıklardan doğrudan etkilenir ve FCA-MFFs ile elde edilen ağırlıklar tahmin olduğundan, beklenen göçmen sayısı farklı olsaydı rotaların ne kadar değişeceğini görmek değerli olacaktır. Testlerimizde, sıcak noktaların ağırlıklarını makul aralıklarda (orijinal değerlerin \pm %15'i) değiştiriyoruz ve sonuçları orijinal çözümlerle karşılaştırıyoruz. Çizelge 6.36 yeni ağırlıkları, Çizelge 6.37 yeni ağırlıklarla elde edilen çözümleri göstermektedir.

Sonuçların yeni ağırlıklarla analizi, orijinal sonuçlardan önemli ölçüde farklı olmadığını bize gösteriyor. Şekil 6.23, ağırlıklar değiştikten sonra limanların yeni etkin sınırlarını göstermektedir; Şekil 6.22'deki etkin sınırlara yakın olduklarını gözlemleyebiliriz. Projemiz için sonuçların ağırlıklardaki makul değişikliklere karşı oldukça sağlam olduğunu söyleyebiliriz.

Çizelge 6.36 : Duyarlılık analizi için yeni ağırlıklar.

Sıcak Nokta No	Eski q_i	Yeni q_i	Sıcak Nokta No	Eski q_i	Yeni q_i	Sıcak Nokta No	Eski q_i	Yeni q_i	Sıcak Nokta No	Eski q_i	Yeni q_i
1	3,30	2,80	26	19,62	22,57	51	2,11	1,79	76	2,79	3,20
2	2,05	2,33	27	6,98	5,93	52	3,57	4,11	77	4,80	4,08
3	5,70	4,84	28	6,50	7,47	53	0,74	0,63	78	5,10	5,87
4	6,73	7,73	29	2,20	1,87	54	4,24	4,88	79	0,26	0,22
5	3,58	3,04	30	5,24	6,02	55	0,83	0,70	80	1,25	1,44
6	9,77	11,24	31	2,45	2,08	56	8,57	9,85	81	1,80	1,53
7	3,10	2,64	32	4,05	4,66	57	4,45	3,78	82	2,57	2,96
8	4,57	5,26	33	1,47	1,25	58	9,66	11,11	83	5,28	4,49
9	3,99	3,39	34	11,95	13,74	59	9,45	8,03	84	1,86	2,14
10	4,68	5,41	35	5,86	4,98	60	1,54	1,77	85	4,08	3,47
11	2,25	1,91	36	1,54	1,77	61	5,99	5,09	86	2,70	3,11
12	7,02	8,07	37	1,16	0,99	62	0,46	0,53	87	4,47	3,80
13	9,87	8,39	38	3,00	3,45	63	4,60	3,91	88	0,14	0,16
14	1,60	1,84	39	0,82	0,70	64	4,81	5,53	89	3,68	3,13
15	3,73	3,17	40	6,17	7,09	65	3,54	3,01	90	5,60	6,44
16	6,77	7,79	41	2,56	2,18	66	1,54	1,77	91	4,20	3,57
17	0,90	0,76	42	5,05	5,81	67	1,54	1,31	92	1,90	2,19
18	3,08	3,54	43	0,56	0,48	68	3,69	4,24	93	0,37	0,31
19	2,57	2,19	44	3,51	4,04	69	4,13	3,51	94	2,25	2,58
20	1,62	1,86	45	3,96	3,37	70	2,36	2,72	95	2,61	2,22
21	2,61	2,22	46	9,58	11,02	71	4,97	4,23	96	1,94	2,23
22	1,86	2,14	47	3,30	2,80	72	5,88	6,77	97	2,08	1,76
23	2,92	2,48	48	2,63	3,03	73	3,74	3,18	98	1,61	1,85
24	1,44	1,66	49	4,56	3,88	74	0,41	0,48	99	5,36	4,56
25	1,70	1,45	50	1,92	2,21	75	6,49	5,51	100	1,96	2,25



Şekil 6.23 : Yeni ağırlıklarla elde edilen etkin çözümler.

Çizelge 6.37 : Yeni ağırlıklarla elde edilen ağırlıklı rotalama çözümleri

Limán	Tespit Gücü	Uzaklık (km)	Atanan araç(lar)
Limán-1	44	518,10	1
	40	510,14	1
	35	475,59	1
	30	405,78	1
	25	279,96	1
	20	260,7	1
	15	195,80	1
	10	167,00	1
	5	152,02	1
	0	0,00	-
Limán -2	95	250,78	2, 3
	90	200,50	2, 3
	80	168,91	3
	70	148,12	3
	60	101,25	3
	50	91,30	2 veya 3
	40	68,60	2 veya 3
	30	60,03	2 veya 3
	20	47,01	2 veya 3
	10	31,40	2 veya 3
0	0,00	-	
Limán -3	101	362,48	4, 5
	90	306,51	4, 5
	80	275,92	5
	70	263,11	5
	60	208,24	5
	50	198,30	4 veya 5
	40	177,61	4 veya 5
	30	166,02	4 veya 5
	20	156,01	4 veya 5
	10	65,44	4 veya 5
0	0,00	-	
Limán -4	148	590,60	6
	135	553,01	6
	125	510,33	6
	110	482,24	6
	90	455,62	6
	80	412,31	6
	70	396,08	6
	60	380,77	6
	50	375,13	6
	40	280,20	6 veya 7
	30	276,78	6 veya 7
	20	100,09	6 veya 7
	10	58,05	6 veya 7
	0	0,00	-

6.3.1.5 Genel deęerlendirme ve yorum

Bu ařamada, Trkiye'de bir deniz hattı gvenlięi projesi kapsamında devriye botu planlaması iin bir MFF-I-MVR yaklařımı neriyoruz. nerilen yaklařımı, sıcak nokta olarak belirlenen ve gmenleri deniz hattında yakalama olasılıęının yksek olduęu dřnlen kritik koordinatların belirlenmesi ve bunların devriye botu ile gezilmesinin saęlanması iin bir pilot alana uyguluyoruz.

alıřmayı  ana ařamada ele alıyoruz. Bu ařamalar talep noktalarının belirlenmesi, nceliklendirilmesi ve rotalama ařamalarıdır. Talep noktalarının belirlenmesi ařamasında sıcak noktaların gemiř gzlem verilerinin kmelenmesi ile elde edilmesini saęlıyoruz. nceliklendirilmeleri iin Blm 6.2'de FCA-MFFs ile elde edilen beklenen gmen sayısı tahmin verilerini kullanıyoruz. Rotalama ařamasında, ilgili projenin olası kořulları ve kısıtlamaları ile ilgili olarak geleneksel ara rotalama senaryosu ve nem aęırlıklı rotalama senaryosu olmak zere iki senaryo ele alıyoruz ve zmleri tartıřıyoruz.

7. SONUÇ VE ÖNERİLER

Bu çalışma ile Türkiye'de Sahil Güvenlik Komutanlığı'nın göçmen tespitini daha etkin hale getirmek amacıyla yürütülen bir milli güvenlik projesi kapsamında, deniz hattındaki göçmen sayısı ilk kez tahmin edilir. Böylece, göç gerçekleşmeden haftalık olarak tahmin edilerek, göçe müdahalenin ulusal güvenlik açısından işlerliğinin sağlanması amaçlanmaktadır.

Çalışmada öncelikle gerçek dünya tarihsel gözlem verilerinden oluşan beş farklı veri seti üzerinde geleneksel ve farklı anlayışlara sahip alternatif olan mevcut yedi tahmin yöntemi uygulanır ve COVID-19 nedeniyle 2020 ve 2021 verileri henüz elde edilemediği için haftalık olarak 2019'un ilk yarısı için göçmen sayısı tahminleri elde edilir. Sonuçlar, her tahmin yönteminin farklı performansa sahip olduğunu göstermektedir. Bu, her veri kümesi için tahmin yöntemlerini birleştirerek daha güvenilir sonuçlar elde etmeyi mümkün kılar. Bu nedenle, daha doğru sonuçlar elde etmek ve çalışmanın başarısını artırmak için, mevcut tahmin yöntemlerini birleştiren FCM tabanlı bir yaklaşım olan önerilen FCA-MFFs yöntemi uygulanmaktadır. Sonuçlar, FCA-MFF'lerin diğer mevcut yöntemlerin performansını iyileştirebileceğini göstermektedir. Beş veri kümesinin tümü için RMSE değerleri açısından tahmin için en iyi performanslar FCA-MFF'den elde edilir.

Çalışmanın amacı, bir ulusal güvenlik projesi için haftalık olarak göçmen sayısını tahmin etmek ve buna bağlı devriye planlaması yapmak olsa da, çalışma, ARIMA, SES, HES, ANFIS, ANN, LSTM ve R-T1FF olmak üzere yedi farklı tahmin yönteminin ilk kez gerçek göçmen verileri üzerinde uygulanması bakımından benzersizdir. Ayrıca bu yedi yöntemin bir kombinasyonu olan önerilen FCA-MFFs yöntemi de ilk kez uygulanmakta ve bu yöntemle daha güvenilir sonuçlar elde edilmektedir.

Çalışmanın ikinci aşamasında, göçmen beklentisine bağlı devriye planlaması yapılmaktadır. Bu aşamada farklı tipte ve dolayısıyla farklı menzile sahip devriye araçları ile çalışılır.

İlk olarak, rotalama modelinin talep noktaları belirlenir. Ziyaret edilecek sıcak noktaların farklı kritikliklere sahip olduğu varsayılır ve buna göre kontrol edilir. Bunun için 2016-2019 yılları arasında göçmenlerin tespit edildiği koordinatları analiz edilir. K-Means kümeleme algoritması ile elde edilen küme merkezlerini sıcak noktalar olarak belirlenir.

Ardından her bir liman alanı için tüm sıcak noktaların ziyaret edildiği geleneksel KARP formülasyonu çözülür. Farklı menzillere sahip devriye araçları ile tüm noktaları ziyaret ederek minimum mesafe çözümünü bulunur. Bu senaryo, tüm devriye araçlarının mevcut olduğu durumu gösterir ve soruna genel bir bakış sağlar.

KV'nin tüm sıcak noktaları ziyaret etmesinin gerekmediği veya kullanılabilir devriye araçlarında sınırların olduğu senaryo olan önem ağırlıklı rotalama üzerinde çalışılır. Bu senaryo, uygulamada mevcut imkanlarla yönlendirme yapılmasına izin verir. Bazı kısıtlamaların varlığında, tüm etkin noktalar gezilebilir değildir. KV 'nin zaman veya devriye botu kısıtlamaları olabilir. Böyle bir durumda KV, minimum mesafe ile en yüksek tespit gücünü elde edeceği rotayı seçmek isteyebilir. Hangi etkin noktaların ziyaret edilmesi gerektiğine göre önceliği seçerken kritiklik ağırlıklarını kullanmanızı öneririz.

Bunu başarmak için, FCA-MFFs ile haftalık olarak 2019'un ilk yarısında 100 sıcak nokta için tahmini göçmen sayısını tahmin ederiz. FCA-MFFs ile tahminler elde etmek için önce basitten karmaşığa doğru 7 farklı tahmin yöntemi ARIMA, SES, HES, ANFIS, ANN, LSTM ve R-T1FF uygularız. Bu sonuçları önerilen FCA-MFFs yaklaşımı yardımıyla birleştirerek daha düşük RMSE'ye sahip ve daha güvenilir tahmin sonuçları elde ederiz. Her hafta için elde edilen tahmin sonuçlarını küme merkezlerini oluşturan koordinatların geçmiş göçmen hikayesine orantılı olarak dağıtarak sıcak noktaların önem ağırlıklarını elde ederiz. Böylece önem ağırlıklarının haftalık olarak beklenen göçmen sayısı ile orantılı olarak güncellenmesini sağlarız.

Yönlendirme probleminin kısıtı olarak devriye araçlarının menzilini kullanırız. Menzil kısıtlamaları altında, bu modeller, sınırlamalara uyan maksimum önem ağırlıklı yönlendirme çözümleri sağlayan çözümler sunar.

Tespit gücünün bir kriter olarak değerlendirilebileceği durumu da inceleriz. KV, kabul edeceği tespit gücünü en üst düzeye çıkarmak ve aynı modelde kat edilen toplam

mesafeyi en aza indirmek isteyebilir. Kriterlerin deęişen seviyelerinde KV'ye alternatif çözümler sunmak için verimli sınırı oluştururuz. Bu verimli sınırda, baskın olmayan, uygulama için aday çözümler olan çözümler bulunur.

Ayrıca, kabul edilebilir tespit gücü seviyelerinin zamanla arttığı durumları da inceliyoruz. Her durum için yönlendirme çözümleri dahil olası yaklaşımları tartışırız. Son olarak, önem ağırlıklı kriterlerde kullanılan sıcak noktaların önem ağırlıkları üzerinde bir duyarlılık analizi gerçekleştiririz. Ağırlıklardaki makul deęişikliklerin sonuçlarda önemli deęişikliklere neden olmayacağından emin olmak istiyoruz. Analizlerimizin sonuçları bu beklentiyi doğrulamaktadır.

Ek olarak mevcut durumda Sahil güvenlik Komutanlığı'nın göçmen tespitine yönelik kullandığı ekipmanların maliyeti ve düzensiz devriye maliyeti ile sunulan devriye planlamasının maliyeti kıyaslandığında önerilen planlama maliyetinin oldukça etkin olduğunu gözlemleriz.

Önerilen yaklaşım ve bu uygulama Türkiye'de bir ilktir. Uygulama oldukça geneldir ve dięer yönlendirme problemlerine kolaylıkla uygulanabilir. Göz önünde bulundurulan kriterler ve kaynaklar üzerindeki kısıtlamalar kolaylıkla deęiştirilebilir. Polis devriye güzergahı, acil durum devriye güzergahı vb. ile ilgili sorun, yaklaşımımızdan ve tartışmalarımızdan yararlanabilir.

Gelecekteki potansiyel çalışma olarak, Zaman Pencereli Meta Bulanık Fonksiyon Entegreli Karma Araç Rotalama (TW-MFF-I-MVR) yaklaşımı geliştirilebilir. Böylece 7/24 devriyeye dayalı olsalar da, KV'nin uygun gördüğü zamanlarda yönlendirme yapmasına olanak sağlayan bir yaklaşım sunulabilir.

Gelecekteki başka bir çalışma olarak, etkin sınırdan en çok tercih edilen çözümü seçmede KV'ye rehberlik edecek yaklaşımlar ele alınabilir. KV'ye bu kriterlerin her ikisinde de bir çözüm seçme konusunda yardımcı olunabilir.

Yine gelecekteki başka bir çalışma olarak, önerilen devriye planlamasının verimliliğini ölçmek için simülasyon teknikleri yardımıyla rassal olarak üretilen göçmenler ve önerilen devriyenin talep noktaları olan sıcak noktaların bu göçmenlere denk gelme durumları karşılaştırılabilir.

KAYNAKLAR

- Abel, G., Bijak, J., Findlay, A., McCollum, D. and Wisniowski, A.** (2013). Forecasting environmental migration to the United Kingdom: an exploration using Bayesian models. *Population and Environment*, 35(2), 183-203.
- Adhikari, R., Agrawal, R.K.** (2013). An Introductory Study on Time Series Modeling and Forecasting.
- Ahmad, A.S., Hassan, M.Y., Abdullah, M.P., Rahman, H.A., Hussin, F., Abdullah, H., Saidur, R.** (2014). A review on applications of ANN and SVM for building electrical energy consumption forecasting. *Renewable and Sustainable Energy Reviews*, 33, 102-109.
- Akın, K.Y.** (2008). Veri Madenciliğinde Kümeleme Algoritmaları ve Kümeleme Analizi. *Doktora Tezi*, Marmara Üniversitesi, Sosyal Bilimler Enstitüsü, İstanbul.
- Aladag, C.H., Turksen, I.B., Dalar, A.Z., Egrioglu, E., Yolcu, U.** (2014). Application of Type-1 Fuzzy Functions Approach for Time Series Forecasting, *Turkish Journal of Fuzzy Systems*, 5(1):1-9.
- Alho, J., Alders, M., Crujisen, H., Keilman, N., Nikander, T., Pham, D. Q.** (2006). New forecast: Population decline postponed in Europe, *Statistical Journal of the United Nations Economic Commission for Europe*, 23, 1-10.
- Arslan, H.** (2008). Sakarya Üniversitesi Web Sitesi Erişim Kayıtlarının Web Madenciliği İle Analizi. *Yüksek Lisans Tezi*, Sakarya Üniversitesi, Fen Bilimleri Enstitüsü, Sakarya.
- Atmaca, E.** (2012). Bir Kargo Şirketinde Araç Rotalama Problemi ve Uygulaması. *TÜBAV Bilim*, 5(2), 12-27.
- Azi, N., Gendreau, M., Potvin, J-Y.** (2010). An exact algorithm for a vehicle routing problem with time windows and multiple use of vehicles. *European Journal of Operational Research*, 202, 756-763.
- Barak, S., Sadegh, S.S.** (2016). Forecasting energy consumption using ensemble ARIMA–ANFIS hybrid algorithm. *International Journal of Electrical Power & Energy Systems*, 82, 92-104.
- Belmecheri, F., Prins, C., Yalaoui, F.** (2013). Particle swarm optimization algorithm for a vehicle routing problem with heterogeneous fleet, mixed backhauls, and time windows. *J Intell Manuf*, 24, 775–789.
- Beyhan, S., Alici, M.** (2010). Fuzzy Functions Based ARX Model and New Fuzzy Basis Function Models for Nonlinear System Identification. *Applied Soft Computing*, 10, 439-444.

- Bezdek, C.** (1981). Pattern recognition with fuzzy objective Junction algorithms. *Plenum Press*.
- Bijak, J.** (2006). Forecasting International Migration: Selected Theories, Models, and Methods. *Central European Forum for Migration Research*, 4, Poland.
- Bijak, J.** (2010). Forecasting International Migration in Europe: A Bayesian View. *Springer Series on Demographic Methods and Population Analysis*, 24.
- Bijak, J., Wisniowski, A.** (2010). Bayesian forecasting of immigration to selected European countries by using expert knowledge. *Journal of the Royal Statistical Society*, 173(4), 775–796.
- Box, G.E.P., Jenkins, G.M.** (1970). Time Series Analysis: Forecasting and Control. Holden-Day, San Francisco.
- Braekers, K., Ramaekers, K., Nieuwenhuysse, I.V.** (2016). The vehicle routing problem: State of the art classification and review. *Computers & Industrial Engineering*, 99, 300-313.
- Brockwell, P. J. and Davis, R.A.** (2002). Introduction to Time Series and Forecasting, *Second Edition*, Springer, Newyork.
- Brownlee, J.** (2016). Time Series Prediction with LSTM Recurrent Neural Networks.
- BuHamra, N. Smaoui, M.G.** (2003). The Box–Jenkins analysis and neural networks: prediction and time series modeling, *Applied Mathematical Modelling*, 27, 805–815.
- Cappelen, A., Skjerpen, T., Tonnessen, M.** (2015). Forecasting Immigration in Official Population Projections Using an Econometric Model. *International Migration Review*, 49(4).
- Cardoso, F.A., Abrahao, F.T.M., Saba, W.B.** (2021). Resources Package Modelling Supporting Border Surveillance Operations. *Defence Science Journal*, 71(3), 410-417.
- Cattaruzza, D., Absi, N., Feillet, D.** (2016). Vehicle routing problems with multiple trips. *4OR-Q J Oper Res*, 14, 223–259.
- Celikyilmaz, A., Turksen, I.B.** (2009). Modeling Uncertainty with Fuzzy Logic: with Recent Theory and Applications, *Springer*, Berlin.
- Chan K.P., Cheung Y.S.** (1992). Clustering of clusters. *Pattern Recognit.*, 25 (2), 211-217.
- Chawathe, S.S.** (2007). Organizing Hot-Spot Police Patrol Routes. *IEEE Intelligence and Security Informatics*, 79–86.
- Chen, H. Chen, T., Wise, S.** (2015). Designing daily patrol routes for policing based on ant colony algorithm. In Proceedings of the ISPRS Annals of the Photogrammetry, *Remote Sensing and Spatial Information Sciences*, 2,103–109.
- Chen, H. Chen, T., Wise, S.** (2017). Developing an online cooperative police patrol routing strategy. *Computers, Environment and Urban Systems*, 62, 19–29.

- Choi, J.Y., Lee, B.** (2018). Combining LSTM Network Ensemble via Adaptive Weighting for Improved Time Series Forecasting. *Hindawi Mathematical Problems in Engineering*, Article ID 2470171.
- Clarke, G., Wright, J.** (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 12, 568–581.
- Cochrane, J.H.** (1997). Time Series for Macroeconomics and Finance. Graduate School of Business, University of Chicago.
- Cordeau, J.F., Laporte, G., Savelsbergh, M.W., Vigo, D.** (2006). Vehicle Routing. Transportation, *Handbooks in Operations Research and Management Science*, 14, 367-428.
- Cottrell, M., Girard, B., Girard, Y., Mangeas, M., Muller, C.** (1995). Neural modeling for time series: a statistical stepwise method for weight elimination. *IEEE Transactions on Neural Networks*, 6(6), 1355-1364
- Dantzig G.B., Ramser J.T.** (1959). The truck dispatching problem, *Management Science*, 6(1), 80-91.
- Dersimonian, R., Laird, N.** (1986). Meta-analysis in clinical trials, *Controlled Clin. Trials*, 7(3), 177-188.
- Dewinter, M., Vandeviver, C., Beken, T.V., Witlox, F.** (2020). Analysing the police patrol routing problem: A review. *ISPRS International Journal of Geo-Information*, 9(3), 157.
- Disney, G.** (2014). Model-Based Estimates of UK Immigration. *PhD Thesis*, University of Southampton
- Dondo, R., Cerda, J.** (2007). A cluster-based optimization approach for the multi-depot heterogeneous fleet vehicle routing problem with time Windows. *European Journal of Operational Research*, 176(3), 1478-1507.
- Eberhart, R.C., Shi, Y.H.** (2001). Particle swarm optimization: Developments, applications and resources. Proc. *IEEE Congr. Evol. Comput.*, Seoul, Korea, 81–86.
- Eden W.M., Tommy, M.A., Chow, W.S.** (2004). A New Shifting Grid Clustering Algorithm. *Pattern Recognition*, 37(3,), 504.
- Egrioglu, E., Aladag, C.H., Yolcu, U., Basaran, M.A., Uslu, V.R.** (2009). A new hybrid approach based on SARIMA and partial high order bivariate fuzzy time series forecasting model, *Expert Systems with Applications*, 36, 7424–7434.
- Etezadi, T., Beasley, J.E.** (1983). Vehicle fleet composition. *Journal of the Operational Research Society*, 34, 87– 91.
- Faraway, J., Chatfield, C.** (1998). Time series forecasting with neural networks: a comparative study using the airline data. *Applied Statistics*, 47, 231–250.
- Franken, N., Engelbrecht, A.P.** (2005). Particle swarm optimization approaches to coevolve strategies for the iterated prisoner’s dilemma. *IEEE Trans. Evol. Comput.*, 9, 562–579.

- Gendreau, M., Hertz, A., Laporte, G.** (1992). New insertion and post optimization procedures for the traveling salesman problem. *Operations Research*, 40, 1086–1094.
- Gendreau, M., Laporte, G., Musaraganyi, C.H., Taillard, E.** (1999). A tabu search heuristic for the heterogeneous fleet vehicle routing problem. *Computers and Operations Research*, 26, 1153–1173.
- Gers, F.A., Schmidhuber, J., Cummins, F.** (2000). Learning to Forget: Continual Prediction with LSTM. *Neural Computation*, 12(10), 2451- 2471.
- Glass G.V.** (1976). Primary, secondary, and meta-analysis of research. *Educ. Researcher*, 5(10), 3-8
- Golden, B., Assad, A., Levy, L., Gheysens, F.** (1984). The fleet size and mix vehicle routing problem. *Computers and Operations Research*, 11, 49–66.
- Gould, J.** (1969). The size and composition of a road transport fleet. *Operational Research Quarterly*, 20, 81–92.
- Guillain, M.S., Paquay, C., Limbourg, S.** (2021). Time-dependent stochastic vehicle routing problem with random requests: Application to online police patrol management in Brussels. *European Journal of Operational Research*, 292(3), 869-885.
- Haimes, Y.Y., Lasdon, L., Wismer, D.** (1971). On a Bicriteria Formulation of the Problems of the Integrated System Identification and System Optimization. *IEEE Transaction on Systems, Man, and Cybernetics*, 296-297.
- Hipel, K.W., McLeod, A.I.** (1994). Time Series Modelling of Water Resources and Environmental Systems, *Elsevier*, Amsterdam.
- Ho, S.Y., Lin, H.S., Liauh, W.H., Ho, S.J.** (2008). OPSO: Orthogonal particle swarm optimization and its application to task assignment problems,. *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, 38, 288–298.
- Hochreiter, S., Schmidhuber, J.** (1997). Long short-term memory. *Neural Computation*, 9(8), 1735-1780.
- Hopfield, J.J.** (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of the Sciences of the U.S.A.*, 79, 2554-2558.
- Hu, M.J.C.** (1964). Application of the adaline system to weather forecasting. *Master Thesis*, Technical Report 6775-1, Stanford Electronic Laboratories, Stanford, CA.
- Jain, A., Kumar, A.M.** (2007) Hybrid neural network models for hydrologic time series forecasting, *Applied Soft Computing*, 7, 585–592.
- Jain, A., Dubes, R.C.** (1988). Algorithms for Clustering Data. *Prentice-Hall Inc.*
- Jang, J.S.R.** (1993). ANFIS: Adaptive-Network-Based Fuzzy Inference System, *IEEE Transactions on Systems, Man and Cybernetics*, 23(3), 665-685.
- Jiang, J., Kien, M.N, Poh, K.L., Teo, K.M.** (2014). Vehicle routing problem with a heterogeneous fleet and time Windows. *Expert Systems with Applications*, 41(8), 3748-3760.

- Karaboga, D., Kaya, E.** (2020). Estimation of number of foreign visitors with ANFIS by using ABC algorithm. *Soft Computing*, 24, 7579–7591.
- Kennedy, J., Eberhart, R.** (1995). Particle Swarm Optimization, Proceedings of the IEEE International Conference on Neural Networks, IEEE press, Piscataway, 1942-1948.
- Khanmohammadi, S., Adibeig, N., Shanebandy, S.** (2017). An improved overlapping k-means clustering method for medical applications. *Expert Syst. Appl.* 67.
- Khashei, M., Bijari, M.** (2010). An artificial neural network (p, d, q) model for timeseries forecasting. *Expert Systems with Applications*, 37(1), 479-489.
- Koç, Ç., Bektaş, T., Jabali, O., Laporte G.** (2016). Thirty years of heterogeneous vehicle routing. *European Journal of Operational Research*, 249(1), 1-21.
- Kohonen, T.** (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43, 59-69.
- Krohling, R.A., Coelho, L.S.** (2009). Coevolutionary particle swarm optimization using Gaussian distribution for solving constrained optimization problems. *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, 36, 1407–1416.
- Kuo, P.F., Lord, D., Walden, T.D.** (2013). Using geographical information systems to organize police patrol routes effectively by grouping hotspots of crash and crime data. *Journal of Transport Geography*, 30 (1), 138-148.
- Kupiszewski, M.** (2002). The role of international migration in the modelling of population dynamics. *Institute of Geography and Spatial Organisation, Polish Academy of Sciences, Warsaw*
- Lapedes, A., Farber, R.** (1987). Nonlinear signal processing using neural networks: Prediction and system modelling. *IEEE international conference on neural networks*, San Diego, CA, USA.
- Li, X., Kang, Y., Li, F.** (2020). Forecasting with time series imaging. *Expert Systems with Applications*, 160.
- Li, X.D., Engelbrecht, A.P.** (2007). Particle swarm optimization: An introduction and its recent developments. *Proc. Genetic Evol. Comput. Conf.*, 3391–3414.
- Liu, B., Wang, L., Jin, Y.H.** (2004). An effective PSO-based memetic algorithm for flow shop scheduling. *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, 37,18–27.
- Luo, H., Zhang, P., Wang, J., Wang, G., Meng, F.** (2019) Traffic Patrolling Routing Problem with Drones in an Urban Road System. *Sensors*, 19(23), 5164.
- Lutz, W., Goldstein, J.R.** (2004). Introduction: how to deal with uncertainty in population forecasting? *International Statistical Review*, 72(1), 1-4.
- Ma, Q.** (2020). Comparison of ARIMA, ANN and LSTM for Stock Price Prediction. *E3S Web of Conferences* 218, 01026.

- MacQueen, J. B.** (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Symposium on Math, Statistics, and Probability*, 281–297.
- Macrina, G., Laporte, G., Guerriero, F., Pugliese, L.G.D.P.** (2019). An energy-efficient green-vehicle routing problem with mixed vehicle fleet, partial battery recharging and time Windows. *European Journal of Operational Research*, 276(3), 971-982.
- Martineau, J.S.** (2010). Red Flags: A Model for the Early Warning of Refugee Outflows. *Journal of Immigrant & Refugee Studies*, 8(2):135-157.
- Matamoros, A., Fujita, H., Hayashi, T., Meana, H.P.** (2020). Forecasting of COVID19 per regions using ARIMA models and polynomial functions. *Applied Soft Computing Journal*, 96.
- Murray, C.C., Park, W.** (2013). Incorporating Human Factor Considerations in Unmanned Aerial Vehicle Routing. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 43(4), 860 - 874.
- Na, S., Xumin, L., Yong, G.** (2010). Research on k-means Clustering Algorithm: An Improved k-means Clustering Algorithm. *Third International Symposium on Intelligent Information Technology and Security Informatics*.
- Namini, S.S., Tavakoli, N., Namin, A.S.** (2018). A Comparison of ARIMA and LSTM in Forecasting Time Series. *17th IEEE International Conference on Machine Learning and Applications*.
- Ozkan, I., Turksen, I.B.** (2007). Upper and lower values for the level of fuzziness in FCM. *Inform. Sci.*, 177 (23), 5143-5152.
- Pal, N.R., Bezdek, C.** (1995). On cluster validity for the fuzzy c-means model. *IEEE Trans. Fuzzy Syst.*, 3 (3), 370-379.
- Qing, X., Niu, Y.** (2018). Hourly day-ahead solar irradiance prediction using weather forecasts by LSTM. *Energy*, 148, 461-468.
- Quttineh, N.H., Larsson, T., Lundberg, K., Holmberg, K.** (2013). Military aircraft mission planning: a generalized vehicle routing model with synchronization and precedence. *EURO J Transp Logist*, 2, 109–127.
- Raicharoen, T., Lursinsap, C., Sanguanbhoki, P.** (2003). Application of critical support vector machine to time series prediction. *Circuits and Systems*, 5, 741-744.
- Raymer, J., Wisniowski, A.** (2018). Applying and testing a forecasting model for age and sex patterns of immigration and emigration. *A Journal of Demography*, 72(3).
- Raymer, J., Wisniowski, A., Forster, J.J., Smith, P.W.F., Bijak, J.** (2013). Integrated Modelling of European Migration. *Journal of the American Statistical Association*, 108, 801 – 819.
- Reilly, D.L., Cooper, L.N.** (1990). An overview of neural networks: early models to real world systems. In: Zornetzer, S.F., Davis, J.L., Lau, C. (Eds.), *An Introduction to Neural and Electronic Networks*. Academic Press, 227–248.

- Renaud, J., Boctor, F.F.** (2002). A sweep-based algorithm for the fleet size and mix vehicle routing problem. *European Journal of Operational Research*,140(3), 618-628.
- Sak, H., Senior, A., Beaufays, F.** (2014). Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In Proc. interspeech.
- Sassi O., Cherif-Khettaf W.R., Oulamara A.** (2015) Multi-start Iterated Local Search for the Mixed Fleet Vehicle Routing Problem with Heterogenous Electric Vehicles. In: Ochoa G., Chicano F. (eds) *Evolutionary Computation in Combinatorial Optimization*.
- Schermer, D., Moeini, M., Wendt, O.** (2019). A matheuristic for the vehicle routing problem with drones and its variants. *Transportation Research Part C: Emerging Technologies*,106, 166-204.
- Sharda, R., Patil, R.B.** (1992). Connectionist approach to time series prediction: An empirical test. *Journal of Intelligent Manufacturing*, 3, 317-323.
- Surendonk, T., Chircop., P.A.** (2021). On the computational complexity of the patrol boat scheduling problem with complete coverage. *Naval Research Logistics*, 67(4), 289-299.
- Suriani, S., Ibn, A.U., Shaikot, H.M.** (2019). A Predictive Model for the Population Growth of Refugees in Asia: A Multiple Linear Regression Approach. *Journal of Computational and Theoretical Nanoscience*, 16 (3), 1196-1202.
- Taillard, E.** (1993). Parallel iteration search methods for vehicle routing. *Networks*, 23, 661–673.
- Tak, N.** (2016). Recurrent Type-1 Fuzzy Functions Approach For Time Series Forecasting. *Phd Thesis* Department Of Statistics Program Of Statistics.
- Tak, N.** (2018). Meta fuzzy functions: Application of recurrent type-1 fuzzy functions. *Applied Soft Computing*, 73,1-13.
- Tak, N.** (2020). Meta fuzzy index functions. *Communications*, 69(1), 654–667.
- Tak, N.** (2021). Forecast Combination with Meta Possibilistic Fuzzy Functions. *Information Sciences*, 160, 168-182.
- Tak, N., Gök, A.** (2020). Dating currency crises and designing early warning systems: Meta-possibilistic fuzzy index functions. *International Journal of Finance and Economics*, 1–18.
- Tak, N., Tez, M., Evren, A., Egrioglu, E.** (2018). Recurrent Type-1 Fuzzy Functions Approach for Time Series Forecasting. *Applied Intelligence*, 48(1), 68-77.
- Tang, Z., Almeida, C., Fishwick, P.A.** (1991). Time series forecasting using neural networks vs Box-Jenkins methodology. *Simulation*, 57 (5), 303-310.
- Tong, H.** (1983). *Threshold Models in Non-Linear Time Series Analysis*. Springer-Verlag, New York, 1983.
- Toth, P., Vigo, D.** (2014). Vehicle Routing Problems, Methods and Applications. *Mathematical Optimization Society*, 2, 349-453.

- Tseng, M., Yu, H.C., Tzeng, G.H.** (2002) Combining neural network model with seasonaltime series ARIMA model, *Technological Forecasting and Social Change*, 69, 71–87
- Turksen, I.B.** (2008). Fuzzy functions with LSE. *Applied Soft Computing*, 8(3), 1178-1188.
- Uslu, V.R. C.H. Aladag, C.H., Yolcu, U., Egrioglu, E.** (2010). A new hybrid approach for forecasting a seasonal fuzzy time series, *International Symposium Computing Science and Engineering Proceeding Book*, 1152–1158.
- Wang, X., Poikonen, S., Golden, B.** (2016). The vehicle routing problem with drones: Extended models and connections, *Networks*, 70, 34–43.
- Warren M., Pitts W.** (1943). A Logical Calculus of Ideas Immanent in Nervous Activity. *Bulletin of Mathematical Biophysics*. 5(4), 115–133.
- Weigend, A.S., Huberman, B.A., Rumelhart D.E.** (1990). Predicting the future: A connectionist approach. *International Journal of Neural Systems*, 1, 193-209.
- Werbos, P.J.** (1974). Beyond regression: new tools for prediction and analysis in the behavioral sciences. *Ph.D. thesis*, Harvard University.
- Werbos, P.J.** (1988). Generalization of backpropagation with application to a recurrent gas market model. *Neural Networks*, 1, 339-356.
- Wicke, L., Dhimi, M.K., Önköl, D., Belton, I.K.** (2019). Using scenarios to forecast outcomes of a refugee crisis. *International Journal of Forecasting*, In press. Corrected Proof, Available online.
- Willekens, F.** (1994). Monitoring international migration flows in Europe: towards a statistical data base combining data from different sources. *European Journal of Population*, 10, 1–42.
- Williams, R.J., Hinton, G.E., Rumelhart, D.E.** (1986). Learning representations by back-propagating errors. *Nature*, 323, 533–536.
- Wisniowski, A.** (2013). Bayesian Modelling of International Migration with Labour Force Survey Data. *PhD Thesis* Collegium of Economic Analyses, Warsaw.
- Woods, D.G., Harris, F.C.,** (1979). Truck fleet size composition for concrete distribution. *International Journal of Physical Distribution*, 10, 3–14.
- Wu, X., Zhu, X., Wu, G.Q., Ding, W.** (2014). Data mining with big data, *IEEE Trans. Knowl. Data Eng.*
- Yung-Yao, C., Yu-Hsiu, L., Chia-Ching, K., Ming-Han, C., I-Hsuan, Y.** (2019). Design and Implementation of Cloud Analytics-Assisted Smart Power Meters Considering Advanced Artificial Intelligence as Edge Analytics in Demand-Side Management for Smart Homes. *Sensors*. 19(9), 2047.
- Zhang, G.P.** (2003). Time series forecasting using a hybrid ARIMA and neural network model, *Neurocomputing*, 50, 159–175.

Zhang, P.G., B., Patuwo, E., Michael, Y.H. (1998). Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting*, 14(1), 35-62



EKLER

EK 1: BölgeBazındaHaftalıkGöçVerileri



EK 1**Çizelge Ek.1 : Bölge Bazında Haftalık Yakalanan Göçmen Verileri.**

Tarih	Kıta Sahaneliği	Bölge-1	Bölge-2	Bölge-3	Bölge-4
2015-hafta-1	99	28	0	44	27
2015-hafta-2	22	0	22	0	0
2015-hafta-3	468	347	80	0	41
2015-hafta-4	69	0	59	0	10
2015-hafta-5	104	100	4	0	0
2015-hafta-6	129	6	12	59	52
2015-hafta-7	182	40	100	0	42
2015-hafta-8	290	8	104	41	137
2015-hafta-9	254	66	124	64	0
2015-hafta-10	168	0	44	91	33
2015-hafta-11	477	66	0	12	399
2015-hafta-12	612	40	205	227	140
2015-hafta-13	532	57	317	54	104
2015-hafta-14	346	88	128	80	50
2015-hafta-15	302	61	32	68	141
2015-hafta-16	783	227	189	142	225
2015-hafta-17	789	213	143	149	284
2015-hafta-18	1312	396	235	140	541
2015-hafta-19	559	283	166	110	0
2015-hafta-20	1198	203	160	265	570
2015-hafta-21	730	392	203	95	40
2015-hafta-22	1173	743	91	136	203
2015-hafta-23	1296	664	287	135	210
2015-hafta-24	1487	370	377	263	477
2015-hafta-25	1247	180	377	91	599
2015-hafta-26	1410	561	281	201	367
2015-hafta-27	1477	195	341	373	568
2015-hafta-28	2296	336	427	384	1149
2015-hafta-29	2938	240	535	342	1821
2015-hafta-30	4251	483	503	999	2266
2015-hafta-31	4233	389	901	1021	1922
2015-hafta-32	4327	594	1059	1646	1028
2015-hafta-33	4866	1090	900	705	2171
2015-hafta-34	2850	532	681	440	1197
2015-hafta-35	4277	806	964	902	1605
2015-hafta-36	3877	259	1166	944	1508
2015-hafta-37	3573	835	893	260	1585
2015-hafta-38	2751	462	537	334	1418
2015-hafta-39	2298	293	460	388	1157
2015-hafta-40	2731	399	666	371	1295
2015-hafta-41	3382	681	1032	609	1060
2015-hafta-42	4332	443	1057	928	1904
2015-hafta-43	1474	301	495	383	295
2015-hafta-44	3248	422	1118	624	1084
2015-hafta-45	2372	354	626	296	1096
2015-hafta-46	1924	175	517	366	866
2015-hafta-47	2037	327	324	94	1292
2015-hafta-48	1933	177	296	449	1011

Tarih	Kıta Sahanlığı	Bölge-1	Bölge-2	Bölge-3	Bölge-4
2015-hafta-49	2414	286	467	781	880
2015-hafta-50	1888	243	501	750	394
2015-hafta-51	1947	394	339	689	525
2015-hafta-52	1964	243	426	624	671
2016-hafta-1	1309	94	344	612	259
2016-hafta-2	859	277	226	221	135
2016-hafta-3	1865	177	467	561	660
2016-hafta-4	718	42	154	124	398
2016-hafta-5	1323	64	525	323	411
2016-hafta-6	1408	161	285	583	379
2016-hafta-7	2026	225	333	607	861
2016-hafta-8	2740	353	317	1043	1027
2016-hafta-9	2593	220	558	989	826
2016-hafta-10	2924	434	310	1482	698
2016-hafta-11	1463	200	94	627	542
2016-hafta-12	2484	116	217	738	1413
2016-hafta-13	930	93	0	320	517
2016-hafta-14	682	26	46	118	492
2016-hafta-15	317	0	52	58	207
2016-hafta-16	395	12	0	93	290
2016-hafta-17	346	45	61	70	170
2016-hafta-18	135	0	0	0	135
2016-hafta-19	224	64	0	160	0
2016-hafta-20	305	19	57	91	138
2016-hafta-21	302	63	0	161	78
2016-hafta-22	305	8	22	59	216
2016-hafta-23	58	0	0	11	47
2016-hafta-24	202	0	43	106	53
2016-hafta-25	121	10	0	72	39
2016-hafta-26	56	0	0	27	29
2016-hafta-27	149	46	103	0	0
2016-hafta-28	78	26	52	0	0
2016-hafta-29	190	76	71	43	0
2016-hafta-30	228	145	0	0	83
2016-hafta-31	509	245	0	120	144
2016-hafta-32	349	96	13	68	172
2016-hafta-33	181	39	0	93	49
2016-hafta-34	593	302	0	85	206
2016-hafta-35	256	42	1	90	123
2016-hafta-36	533	211	0	105	217
2016-hafta-37	935	129	21	307	478
2016-hafta-38	858	321	8	299	230
2016-hafta-39	822	257	93	194	278
2016-hafta-40	848	15	48	349	436
2016-hafta-41	664	288	201	56	119
2016-hafta-42	350	0	62	201	87
2016-hafta-43	712	246	159	101	206
2016-hafta-44	256	1	123	87	45
2016-hafta-45	519	100	109	129	181
2016-hafta-46	537	50	59	42	386
2016-hafta-47	456	0	217	89	150
2016-hafta-48	235	0	21	43	171
2016-hafta-49	104	48	1	55	0
2016-hafta-50	163	0	0	95	68

Tarih	Kıta Sahanlığı	Bölge-1	Bölge-2	Bölge-3	Bölge-4
2016-hafta-51	131	2	0	0	129
2016-hafta-52	373	359	0	0	14
2017-hafta-1	120	14	0	24	82
2017-hafta-2	102	102	0	0	0
2017-hafta-3	75	59	16	0	0
2017-hafta-4	134	0	1	91	42
2017-hafta-5	301	1	0	0	300
2017-hafta-6	265	56	19	99	91
2017-hafta-7	88	19	69	0	0
2017-hafta-8	135	0	0	21	114
2017-hafta-9	230	41	0	50	139
2017-hafta-10	329	18	54	256	1
2017-hafta-11	274	0	99	174	1
2017-hafta-12	437	104	41	175	117
2017-hafta-13	532	171	46	262	53
2017-hafta-14	321	44	84	193	0
2017-hafta-15	460	18	216	86	140
2017-hafta-16	374	0	0	213	161
2017-hafta-17	387	1	106	231	49
2017-hafta-18	2	0	0	2	0
2017-hafta-19	217	0	105	112	0
2017-hafta-20	347	0	0	229	118
2017-hafta-21	539	166	87	142	144
2017-hafta-22	450	0	137	184	129
2017-hafta-23	135	18	24	93	0
2017-hafta-24	696	254	76	212	154
2017-hafta-25	461	147	171	53	90
2017-hafta-26	497	33	206	108	150
2017-hafta-27	312	57	62	64	129
2017-hafta-28	505	183	159	50	113
2017-hafta-29	423	115	112	121	75
2017-hafta-30	309	0	38	52	219
2017-hafta-31	703	1	263	60	379
2017-hafta-32	526	15	345	35	131
2017-hafta-33	253	1	78	61	113
2017-hafta-34	971	36	129	83	723
2017-hafta-35	743	104	154	305	180
2017-hafta-36	1128	78	210	144	696
2017-hafta-37	677	122	44	86	425
2017-hafta-38	796	83	158	131	424
2017-hafta-39	379	67	60	12	240
2017-hafta-40	627	0	199	290	138
2017-hafta-41	683	117	28	211	327
2017-hafta-42	799	157	209	332	101
2017-hafta-43	689	65	120	275	229
2017-hafta-44	859	127	47	0	685
2017-hafta-45	400	45	19	0	336
2017-hafta-46	402	0	32	201	169
2017-hafta-47	338	101	0	69	168
2017-hafta-48	148	0	0	0	148
2017-hafta-49	478	65	95	190	128
2017-hafta-50	304	43	38	12	211
2017-hafta-51	485	0	41	152	292
2017-hafta-52	165	0	95	7	63

Tarih	Kıta Sahanlığı	Bölge-1	Bölge-2	Bölge-3	Bölge-4
2018-hafta-1	600	20	71	224	285
2018-hafta-2	481	0	85	279	117
2018-hafta-3	154	0	105	0	49
2018-hafta-4	301	0	58	50	193
2018-hafta-5	485	292	27	0	166
2018-hafta-6	417	59	47	157	154
2018-hafta-7	265	29	103	50	83
2018-hafta-8	224	80	0	91	53
2018-hafta-9	397	43	58	58	238
2018-hafta-10	273	1	70	6	196
2018-hafta-11	259	202	1	56	0
2018-hafta-12	153	0	0	0	153
2018-hafta-13	1023	162	240	178	443
2018-hafta-14	612	10	227	99	276
2018-hafta-15	687	48	176	411	52
2018-hafta-16	488	0	38	215	235
2018-hafta-17	516	64	83	329	40
2018-hafta-18	941	184	56	487	214
2018-hafta-19	931	0	128	189	614
2018-hafta-20	1066	54	294	229	489
2018-hafta-21	380	0	67	161	152
2018-hafta-22	460	4	43	197	216
2018-hafta-23	295	23	22	161	89
2018-hafta-24	560	0	63	127	370
2018-hafta-25	458	34	129	177	118
2018-hafta-26	493	15	27	346	105
2018-hafta-27	428	39	93	148	148
2018-hafta-28	287	13	90	184	0
2018-hafta-29	764	110	116	267	271
2018-hafta-30	672	195	143	129	205
2018-hafta-31	270	8	124	46	92
2018-hafta-32	87	46	41	0	0
2018-hafta-33	447	103	52	94	198
2018-hafta-34	511	37	0	247	227
2018-hafta-35	690	141	136	185	228
2018-hafta-36	758	119	58	176	405
2018-hafta-37	515	74	113	15	313
2018-hafta-38	954	92	92	333	437
2018-hafta-39	443	0	51	99	293
2018-hafta-40	902	150	214	311	227
2018-hafta-41	284	39	144	101	0
2018-hafta-42	753	19	154	494	86
2018-hafta-43	913	45	263	330	275
2018-hafta-44	522	88	16	328	90
2018-hafta-45	352	152	54	78	68
2018-hafta-46	184	0	0	93	91
2018-hafta-47	353	24	127	33	169
2018-hafta-48	590	29	152	121	288
2018-hafta-49	284	0	219	8	57
2018-hafta-50	763	0	445	114	204
2018-hafta-51	536	0	242	88	206
2018-hafta-52	403	17	56	149	181
2019-hafta-1	173	0	94	79	0
2019-hafta-2	409	0	256	21	132

Tarih	Kıta Sahanlığı	Bölge-1	Bölge-2	Bölge-3	Bölge-4
2019-hafta-3	388	0	77	64	247
2019-hafta-4	122	0	43	40	39
2019-hafta-5	427	0	274	5	148
2019-hafta-6	385	0	168	49	168
2019-hafta-7	242	46	0	91	105
2019-hafta-8	244	0	71	106	67
2019-hafta-9	555	20	55	114	366
2019-hafta-10	346	18	82	116	130
2019-hafta-11	392	0	70	183	139
2019-hafta-12	502	68	163	144	127
2019-hafta-13	157	11	47	99	0
2019-hafta-14	758	0	293	105	360
2019-hafta-15	568	21	353	153	41
2019-hafta-16	497	0	54	162	281
2019-hafta-17	864	68	304	143	349
2019-hafta-18	87	6	52	0	29
2019-hafta-19	773	7	502	184	80
2019-hafta-20	715	100	354	149	112
2019-hafta-21	499	53	102	130	214
2019-hafta-22	580	51	252	213	64
2019-hafta-23	488	18	213	186	71
2019-hafta-24	749	0	101	228	420
2019-hafta-25	1034	0	257	281	496
2019-hafta-26	524	0	130	29	365
2019-hafta-27	868	10	107	284	467
2019-hafta-28	1011	44	42	246	679
2019-hafta-29	712	63	307	15	327
2019-hafta-30	1248	185	249	245	569
2019-hafta-31	1444	129	275	411	629
2019-hafta-32	1267	32	393	162	680
2019-hafta-33	1969	59	236	498	1176
2019-hafta-34	1794	83	253	310	1148

