



**TRAFİK UYGULAMALARINDA
VERİ ANALİTİĞİ VE OPTİMİZASYON**
Doktora Tezi

Zeynep İdil ERZURUM ÇİÇEK

Eskişehir 2020

TRAFİK UYGULAMALARINDA VERİ ANALİTİĞİ VE OPTİMİZASYON

Zeynep İdil ERZURUM ÇİÇEK

Doktora Tezi

Endüstri Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Zehra KAMIŞLI ÖZTÜRK

Eskişehir

Eskişehir Teknik Üniversitesi

Lisansüstü Eğitim Enstitüsü

Ocak 2020

Bu tez çalışması BAP Komisyonu tarafından kabul edilen 1709F506 no.lu proje kapsamında desteklenmiştir.

JÜRİ VE ENSTİTÜ ONAYI

Zeynep İdil ERZURUM ÇİÇEK'in TRAFİK UYGULAMALARINDA VERİ ANALİTİĞİ VE OPTİMİZASYON başlıklı tezi 21/01/2020 tarihinde aşağıdaki jüri tarafından değerlendirilerek "Eskişehir Teknik Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliği"nin ilgili maddeleri uyarınca, Endüstri Mühendisliği Anabilim dalında Doktora tezi olarak kabul edilmiştir.

<u>Jüri Üyeleri</u>	<u>Unvan Adı Soyadı</u>	<u>İmza</u>
Üye (Tez Danışmanı)	: Doç. Dr. Zehra KAMIŞLI ÖZTÜRK	
Üye	: Prof. Dr. Onur KAYA	
Üye	: Prof. Dr. Bülent ÇATAY	
Üye	: Doç. Dr. Gürkan ÖZTÜRK	
Üye	: Dr. Öğr. Üyesi Sinan AYDIN	

Prof. Dr. Murat TANIŞLI

Lisansüstü Eğitim Enstitüsü Müdürü

ÖZET

TRAFİK UYGULAMALARINDA VERİ ANALİTİĞİ VE OPTİMİZASYON

Zeynep İdil ERZURUM ÇİÇEK

Endüstri Mühendisliği Anabilim Dalı

Eskişehir Teknik Üniversitesi, Lisansüstü Eğitim Enstitüsü, Ocak 2020

Danışman: Doç. Dr. Zehra KAMIŞLI ÖZTÜRK

Ulaşım, toplumdaki tüm insanların vazgeçilmez ihtiyacıdır. Yaya ya da araç ile şehir içi ya da şehir dışı seyahat etmek, nüfus ve araç sayısı artışı sebebiyle gün geçtikçe zorlaşmaktadır. Ulaşım sebebiyle kaybedilen zamanın yol açtığı maddi ve manevi maliyet katlanılması güç bir hale gelmektedir. Trafik sıkışıklığı, ihlaller ve kazalar sebebiyle ortaya çıkan trafik stresi, birçok insanın iş ve ev yaşantısına da olumsuz bir zemin oluşturmaktadır; hatta trafik stresinin çalışanların verimliliğini düşürdüğü bile düşünülmektedir. Trafikte yaşanan zaman kaybı sebebiyle ortaya çıkan maliyetin boyutu ise oldukça yüksektir. Bu zararlardan korunmak ve şiddetlerini azaltmak için trafikte optimal rotaların seçilmesi, trafik darboğazlarının belirlenmesi, performans ölçümü, kaza analizi vb. konularda optimizasyon, simülasyon ve tahmin çalışmaları yapılmaktadır.

Bu çalışmada trafik ile ilgili yaşanan sorunlar için veri analitiği temelli çalışmalar yapılmıştır. Çalışmada bir zaman serisi olarak ele alınan trafik akışının tahminlemesi için Yapay Sinir Ağı temelli özgün ve melez bir model önerilmiştir. Önerilen melez algoritma çok bilinen zaman serisi tahminleme yöntemleri ile karşılaştırıldığında oldukça başarılı tahminler elde edildiği görülmüştür. Bunun yanı sıra, Eskişehir’de yaşanan kazaların analizi yapılmış, olasılık dağılımları belirlenmiş ve kaza tahmini için veri madenciliği yöntemlerinden faydalanılmıştır. Tahminlerden ve elde edilen olasılık dağılımlarından faydalanarak, belirsiz ve dinamik olan trafik ağlarında en kısa süren rotayı çevrimdışı olarak belirlemek için Markov Karar Süreci temelli bir dinamik programlama yaklaşımı uygulanmıştır. Uygulanan bütünsel yaklaşım, üretilen problemler üzerinden test edildiğinde deterministik yaklaşıma göre daha iyi sonuçlar elde edilmiştir.

Anahtar Sözcükler: Trafik, Tahminleme, Yapay sinir ağları, Markov karar süreci.

ABSTRACT

DATA ANALYTICS AND OPTIMIZATIN IN TRAFFIC APPLICATIONS

Zeynep İdil ERZURUM ÇİÇEK

Department of Industrial Engineering

Eskişehir Technical University, Institute of Graduate Programs, January 2020

Supervisor: Assoc. Prof. Dr. Zehra KAMIŞLI ÖZTÜRK

Transportation is an indispensable need of all people in society. It is becoming more and more difficult to travel by foot or by car in or out of the city due to the increase in population and number of vehicles. The material and moral costs caused by the time lost due to transportation become difficult to bear. Traffic stress due to traffic congestion, violations and accidents also create a negative ground for many people's business and domestic life; even traffic stress is thought to reduce employee productivity. Therefore, the size of the cost due to the loss of time in traffic is quite high. To prevent and reduce the severity of these damages, optimization, simulation, and prediction studies are carried out for finding optimal routes in traffic, identifying traffic bottlenecks, analyzing accidents, etc.

In this study, data analytical studies are conducted for the problems related to traffic. Firstly, a novel and hybrid model based on Artificial Neural Network is proposed for forecasting the traffic flow which is considered as a time series. Compared with the well-known time series forecasting methods, the proposed hybrid algorithm is found to be more successful. In addition, the accidents in Eskişehir are analyzed, probability distributions of these accidents are determined and data mining methods are used for accident prediction. Lastly, a dynamic programming approach based on the Markov Decision Process is used to determine the shortest route offline in uncertain and dynamic traffic networks by making use of forecasts and obtained probability distributions. When the applied holistic approach is tested on the generated problems, better results are obtained than the deterministic approach.

Keywords: Traffic, Forecasting, Artificial neural networks, Markov decision process.

TEŞEKKÜR

Öncelikle bu çalışmada bana yol gösteren kıymetli hocam Doç. Dr. Zehra KAMIŞLI ÖZTÜRK'e desteklerinden dolayı teşekkür ederim. Fikirleri ve yorumlarıyla çalışmama katkıda bulunan değerli tez izleme jürim Doç. Dr. Gürkan ÖZTÜRK ve Dr. Öğretim Üyesi Sinan AYDIN'a teşekkürlerimi sunarım. Çalışmama katkıda bulunan Prof. Dr. Onur KAYA'ya teşekkürü bir borç bilirim.

Ayrıca, 1709F506 nolu doktora tez projemi destekleyen Eskişehir Teknik Üniversitesi Bilimsel Araştırma Projeleri Komisyonu'na; çalışmalarım için veri sağlayan Eskişehir Büyükşehir Belediyesi Trafik Kontrol Merkezi'nin değerli çalışanlarına teşekkür ederim.

Akademik çalışmalarımda benimle bilgilerini paylaşıp, her zaman destek veren değerli yol arkadaşım Araş. Gör. Zeliha ERGÜL AYDIN'a teşekkürlerimi sunarım.

Son olarak, her zaman desteğini hissettiğim sevgili annem Aysun ERZURUM'a ve bu zorlu ve yoğun süreçte destek veren sevgili eşim Lütfü ÇİÇEK'e sonsuz teşekkürlerimi sunarım.

Zeynep İdil ERZURUM ÇİÇEK

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Bu tezin bana ait, özgün bir çalışma olduğunu; çalışmamın hazırlık, veri toplama, analiz ve bilgilerin sunumu olmak üzere tüm aşamalarında bilimsel etik ve kurallara uygun davrandığımı; bu çalışma kapsamında elde edilen tüm veri ve bilgiler için kaynak gösterdiğimi ve bu kaynaklara kaynakçada yer verdiğimi; bu çalışmanın Eskişehir Teknik Üniversitesi tarafından kullanılan “bilimsel intihal tespit programı”yla tarandığını ve hiçbir şekilde “intihal içermediğini” beyan ederim. Herhangi bir zamanda, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun saptanması durumunda, ortaya çıkacak tüm ahlaki ve hukuki sonuçları kabul ettiğimi bildiririm.

Zeynep İdil ERZURUM ÇİÇEK

İÇİNDEKİLER

Sayfa

BAŞLIK SAYFASI.....	i
JÜRİ VE ENSTİTÜ ONAYI.....	ii
ÖZET	iii
ABSTRACT	iv
TEŞEKKÜR	v
ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ	vi
İÇİNDEKİLER	vii
TABLolar DİZİNİ.....	x
ŞEKİLLER DİZİNİ	xii
SİMGELER VE KISALTMALAR DİZİNİ	xiv
1. GİRİŞ.....	1
1.1. Veri Analitikleri	2
1.2. İçerik ve Ana Katkı.....	4
1.3. Ele Alınan Yol Ağı.....	5
2. TRAFİK AKIŞ TAHMİNLEMESİ.....	8
2.1. Zaman Serileri	8
2.2. Zaman Serileri Tahmin Yöntemleri	10
2.2.1. Geleneksel yöntemler.....	10
2.2.1.1. Otoregresif birleştirilmiş hareketli ortalamalar yöntemi	11
2.2.2. Makine öğrenmesi yöntemleri	12
2.2.2.1. Destek vektör regresyon.....	13
2.2.2.2. Yapay sinir ağları	14
2.3. Literatür Taraması.....	17
2.4. Trafik Akış Tahminlemesi İçin Melez Bir Algoritma Önerisi	18
2.4.1. Yanlı rastgele anahtarlı genetik algoritmalar.....	21
2.4.1.1. Başlangıç çözümü.....	22
2.4.1.2. Çözücü.....	22

2.4.1.3. Seçim ve yeni jenerasyonun oluşturulması.....	23
2.4.1.4. Mutasyon.....	23
2.4.1.5. Çaprazlama.....	24
2.4.2. Trafik akış tahminlemesi için yanlış rastgele anahtarlı genetik algoritma temelli yapay sinir ağı algoritması.....	24
2.4.3. Uygulama ve veri setleri	29
2.4.4. Algoritma parametrelerinin seçimi.....	34
2.4.5. Hesaplamalı sonuçlar	35
2.5. Sonuç.....	40
3. TRAFİK KAZALARININ ANALİZİ VE TAHMİNİ.....	41
3.1. Literatür Taraması.....	41
3.2. Tahminleme Yöntemleri.....	44
3.2.1. Rastgele orman algoritması.....	44
3.2.2. K-en yakın komşuluk yöntemi.....	45
3.2.3. Lojistik regresyon.....	45
3.2.4. Destek vektör makineleri.....	46
3.2.5. Tek sınıf destek vektör makineleri.....	46
3.3. Kaza Verileri.....	47
3.3.1. Veri ön işleme	50
3.3.2. Temel istatistikler	51
3.4. Kaza Gerçekleşme Olasılıklarının Belirlenmesi	53
3.5. Öznitelik Seçimi.....	55
3.6. Eskişehir’de Gerçekleşen Kazaların Kümelenmesi.....	59
3.7. Eskişehir İçin Trafik Kaza Tahmini.....	62
3.7.1. Ters örnek oluşturma	62
3.7.2. Hesaplamalı sonuçlar	63
3.8. Sonuç.....	66
4. DİNAMİK VE STOKASTİK TRAFİK AĞLARINDA ROTA ÖNERİSİ	68
4.1. Literatür Taraması	69
4.2. En Kısa Yol Problemleri	70
4.3. Markov Karar Süreci.....	72

4.4. Dinamik ve Stokastik Ağlarda En Kısa Yol Problemi için Dinamik Programlama Temelli Bir Çözüm Yaklaşımı	73
4.4.1. Durum uzayı	78
4.4.2. Aksiyon uzayı	79
4.4.3. Amaç fonksiyonu	79
4.4.4. Geçiş matrisi	80
4.4.5. Çözüm yaklaşımı	81
4.4.6. Senaryolar	82
4.4.7. Test problemleri.....	83
4.4.8. Hesaplamalı sonuçlar	85
4.5. Sonuç.....	89
5. SONUÇLAR VE TARTIŞMA	91
KAYNAKÇA	93

TABLÖLAR DİZİNİ

Sayfa

Tablo 2.1. YSA optimizasyonu ile ilgili çalışmalar	20
Tablo 2.2. YRAGA-YSa algoritması adımları.....	29
Tablo 2.3. Seçilen zaman serisi veri setleri.....	30
Tablo 2.4. Trafik akış veri setleri	32
Tablo 2.5. Zaman serisi tahminleme veri setinin oluşturulması	33
Tablo 2.6. Parametre değerleri	34
Tablo 2.7. YRAGA-YSa algoritması içi seçilen parametreler.....	34
Tablo 2.8. Seçilen verisetleri için tahminleme sonuçları	36
Tablo 2.9. MAPE değerleri ölçeği.....	36
Tablo 2.10. Wilcoxon işaret testi sonucu karşılaştırmalara ait elde edilen p- değerleri.....	38
Tablo 2.11. Trafik akış verisetleri için tahminleme sonuçlar.....	38
Tablo 3.1. Kaza öznitelikleri.....	49
Tablo 3.2. Kümeleme analizi sonucu oluşturulan kümeler ve içerdikleri kaza sayıları	60
Tablo 3.3. Değişebilir öznitelikler.....	62
Tablo 3.4. Ters örnek oluşturma prosedürü	63
Tablo 3.5. Model parametre aralıkları	64
Tablo 3.6. Veri seti büyüklükleri.....	64
Tablo 3.7. Testler sonucu elde edilen F-skorları	65
Tablo 4.1. Dijkstra algoritması.....	71
Tablo 4.2. Trafiğin aksama seviyeleri	74
Tablo 4.3. Döğüm ekleme ve silme işlemleri	77
Tablo 4.4. Çözüm yaklaşımı adımları.....	82
Tablo 4.5. Senaryo bilgileri.....	83
Tablo 4.6. Eskişehir test problem için oluşturulan senaryolara örnekler.....	84
Tablo 4.7. Eskişehir örnek problemi için elde edilen sonuçlar	85
Tablo 4.8. Senaryo 1 için test problemlerine ait elde edilen ortalama seyahat süreleri (dakika)	86
Tablo 4.9. Senaryo 2 için test problemlerine ait elde edilen ortalama seyahat süreleri (dakika)	87

Tablo 4.10. Senaryo 3 için test problemlerine ait elde edilen ortalama seyahat süreleri (dakika)	88
Tablo 4.11. Elde edilen ortalama ortalama seyahat süreleri (dakika).....	89



ŞEKİLLER DİZİNİ

Sayfa

Şekil 1.1. Eskişehir yol ağı	6
Şekil 1.2. Bir kavşağa ait kamera görüntüsü	6
Şekil 2.1. İki günlük trafik akışı	8
Şekil 2.2. Mevsimsellik, trend ve döngü içeren zaman serileri için örnekler	9
Şekil 2.3. Zaman serisi çizelgesi.....	9
Şekil 2.4. Üç katmanlı bir YSA örneği	14
Şekil 2.5. Sigmoid, tanh ve ReLU aktivasyon fonksiyonları	16
Şekil 2.6. Yanlı rastgele anahtarlı genetik algoritmanın akış şeması.....	22
Şekil 2.7. Yanlı rastgele anahtarlı genetik algortmada seçim ve yeni jenerasyonun oluşturulması.....	23
Şekil 2.8. Yanlı rastgele anahtarlı genetik algortmada çaprazlama süreci	24
Şekil 2.9. Üç gizli nöron ve 2 girdi nöronuna sahip üç katmanlı bir YSA ve bu mimariye ait kromozom örneği.....	26
Şekil 2.10. Kromozomun çözümlenmesi	27
Şekil 2.11. Test edilen veri setleri.....	31
Şekil 2.12. Trafik akış veri setleri	32
Şekil 2.13. Test verisetleri için gerçek ve tahmin değerleri	37
Şekil 2.14. Trafik akış verisetleri için gerçek ve tahmin değerleri	39
Şekil 2.15. Eğitim ve doğrulama verisetlerine ait iterasyon bazında uyum fonksiyonu değerleri.....	40
Şekil 3.1. Türkiye’de gerçekleşen kazalar ve ölü-yaralı sayıları.....	41
Şekil 3.2. DVM’lerin çalışma prensibi.....	46
Şekil 3.3. Eskişehir’de gerçekleşen kaza sayıları	48
Şekil 3.4. Önışleme adımları	50
Şekil 3.5. Kategorik değişkenlerin sayısallaştırılması	50
Şekil 3.6. Ölümlü-yaralanmalı kazalara ait özniteliklerin korelasyon matrisi	51
Şekil 3.7. Maddi hasarlı kazalara ait özniteliklerin korelasyon matrisi	52
Şekil 3.8. Kazaya sebebiyet veren sürücü kusurları.....	52
Şekil 3.9. Kazaların frekans dağılımları.....	53
Şekil 3.10. Belirlenen olasılık dağılımlarına göre beklenen ve gerçekleşen değerler	55

Şekil 3.11. Ölümlü-yaralanmalı kazaların öznelik önem dereceleri.....	57
Şekil 3.12. Ölümlü-yaralanmalı kazaların öznelik birikimli önem dereceleri	57
Şekil 3.13. Maddi hasarlı kazaların öznelik önem dereceleri.....	58
Şekil 3.14. Maddi hasarlı kazaların öznelik birikimli önem dereceleri	58
Şekil 3.15. Ölümlü-yaralanmalı kazalar için doğru ve yanlış alarm oranları.....	66
Şekil 3.16. Maddi hasarlı kazalar için doğru ve yanlış alarm oranları.....	66
Şekil 4.1. Çalışmanın kapsamı.....	73
Şekil 4.2. Aksama seviyelerinde trafik ağının durumu	75
Şekil 4.3. Bir ağ örneği.....	78
Şekil 4.4. Eskişehir test problemi	84
Şekil 4.5. Uygulanan testlerde çözüm yöntemlerinin baskın gelme sayıları.....	89

SİMGELER VE KISALTMALAR DİZİNİ

ARİMA	: Otogregresif Birleştirilmiş Hareketli Ortalamalar
EKY	: En Kısa Yol Problemi
DVM	: Destek Vektör Makineleri (Support Vector Machines)
GA	: Genetik Algoritmalar (Genetic Algorithms)
GPS	: Küresel Konumlama Sistemi (Global Positioning System)
kNN	: K En Yakın Komşu (K-Nearest Neighborhood)
LR	: Lojistik Regresyon (Logistic Regression)
ODVM	: Tek Sınıf Destek Vektör Makineleri (One-class Support Vector Machines)
RAGA	: Rastgele Anahtarlı Genetik Algoritma (Random Key Genetic Algorithms)
ReLU	: Rektifiye Lineer Birimler (Rectified Linear Units)
RO	: Rastgele Ormanlar (Random Forests)
PSO	: Parçacık Sürü Optimizasyonu (Particle Swarm Optimization)
RBF	: Radyal Temelli Fonksiyon (Radial Basis Function)
SARİMA	: Mevsimel Otogregresif Birleştirilmiş Hareketli Ortalamalar
MKS	: Markov Karar Süreci
YRAGA	: Yanlı Rastgele Anahtarlı Genetik Algoritma (Biased Random Key Genetic Algorithms)
YRAGA-	: Yanlı Rastgele Anahtarlı Genetik Algoritma Temelli Yapay Sinir Ağı
YSA	
YSA	: Yapay Sinir Ağları (Artificial Neural Networks)

1. GİRİŞ

Sürücü, yolcu veya yaya olarak sürekli içinde bulunduğumuz ağlar trafik olarak adlandırılır. Trafik dinamik olarak değişen çevre koşullarından anlık olarak etkilenir ve normal akışı kolaylıkla bozulabilir. Trafik ağlarında fazla talep gerçekleştiğinde kapasite – talep dengesizliği nedeniyle sıkışıklıklar meydana gelmekte, trafiğin olağan akışı aksamaktadır. Ayrıca, trafik kazaları, yol çalışmaları, yanlış park vb. sebeplerle de trafik akışında sorunlar meydana gelmektedir. Trafikte yaşanan sorunlar trafik ağının aksaması dışında; aynı zamanda sürücü, yolcu veya yayaların zaman, güvenlik ve sağlık konularında kayıpları ile sonuçlanabilir.

Trafikte beklemekle kaybedilenin sadece süre olduğu düşünülmemelidir. Bekleme süresince tüketilen yakıt, kaybedilen süre esnasında yapılabilecek işlerin yapılamaması, yaşanan trafik stresinden dolayı ortaya çıkan sağlık sorunları vb. sebeplerle trafikte beklemenin maliyeti oldukça yüksektir. Hareketlilik analizi ve bağlantılı araç hizmetleri konusunda çalışan bir firmanın yaptığı 38 ülkede 200’den fazla şehirde tıkanıklık ve hareketlilik eğilimlerini belirleyen ve sıralayan çalışma sonucu sundukları 2018 Küresel Trafik Puan Kartına göre Amerika’da trafik sıkışıklığı nedeniyle sürücü başına yılda ortalama 97 saat süre kaybedilmiş ve bu kayıplar 2018’de yaklaşık 87 milyon dolara mal olmuştur (Reed ve Kidd, 2019). Kayıp maliyet sürücü başına 1348 dolar olarak hesaplanmıştır. Aynı çalışmada oluşturulan Dünya’nın en çok sıkışıklık yaşayan şehirleri sıralamasında İstanbul, Moskova’dan sonra ikinci sırada yer almıştır.

Bunlara ek olarak, trafik sıkışıklıkları trafikteki hareketlilik oranını düşürmesinin yanı sıra; karbon monoksit (CO), karbon dioksit (CO₂), uçucu organik bileşikler (VOC) veya hidrokarbonlar (HC’ler), azot oksitler (NO_x), partikül maddeler (PM) ve araçlarla ilişkili diğer kirleticilerin emisyonlarını artırır. Çeşitli epidemiyolojik çalışmalar, motorlu taşıt emisyonlarına bağlı etkilere odaklanmıştır ve sürücüler, her gün uzun süre yolculuk edenler ile karayoluna yakın yaşayanlar için alerjik olmayan solunum morbiditesi, kardiyovasküler morbidite, kanser, alerjik hastalıklar, olumsuz gebelik ve doğum sonuçları, erkek doğurganlığının azalma risklerini ortaya koymuşlardır (Shashank vd., 2017). Trafik sıkışıklıklarının sürücülerde öfke ve strese sebep olduğu çalışmalarla ortaya konmuştur. Saldırganlık ve nevroz gibi kişilik özelliklerinin, bilgi işleme mekanizmasını bozan ve riskli davranışlara ve güvenlik kaybına yol açan öfkeli bir duygu durumunu ortaya çıkarmak için trafik sıkışıklığı gibi dış stres faktörleriyle etkileşime

girdiğini belirtmişlerdir (Emo, Matthews ve Funke, 2016). Bu sebeple öfke ve stres, trafik güvenlik seviyesini olumsuz yönde etkileyerek trafikte aksama ve kaza olarak geri dönebilmektedir. Ayrıca, stresin uzun dönemde birçok sağlık sorununa sebep olduğu bilinmektedir.

Trafikte yaşanan bu olumsuz durumların ortadan kaldırılması ve trafiğin aksamadan akışının sağlanması konusunda yerel yönetimler çeşitli çözümler bulmaya çalışmaktadır. Bu amaçlarla trafiğin izlenebilirliği ve kontrol edilebilirliğini artırmak için akıllı olarak tabir edilen sistemler kurulmaktadır. Avrupa Birliği, akıllı ulaşım sistemlerini güvenliği artıran ve sıkışıklık ile emisyonu azaltan bir çözüm olarak tanımlamaktadır (İsbak, 2018). Akıllı ulaşım sistemlerinin alt bileşenlerinden biri de gelişmiş trafik yönetim sistemleridir. Gelişmiş trafik yönetim sistemleri tıkanıklığı dinamik olarak yönetir. Akıllı sistemler kullanarak mevcut altyapı kullanım verimliliğini artırır (İsbak, 2018). Gelişmiş trafik sistemleri veri toplama ve işleme, farklı kaynaklardan veri toplam, toplanan verileri birleştirme ve bu verilerden sıkışıklık tahmin etme gibi fonksiyonları bünyesinde barındırır.

Bu çalışmada da trafikte yaşanan sorunların çözümüne yönelik olarak trafik sıkışıklığı ve kazaları ile ilgili veri analitiği uygulamaları yapılmış, tahminlerden elde edilen bilgiler doğrultusunda trafik ağında optimal rota bulmak için bir çözüm yaklaşımı sunulmuştur. Çalışmanın devamında çalışmanın katkısı ve içeriği ile veri analitikleri ve çalışmadaki yeri belirtilmiş; ele alınan trafik ağının özellikleri ve yapısı açıklanmıştır. İkinci bölümde ise trafik akışı tahminlemesi için melez bir algoritma önerilmiştir. Çalışmanın üçüncü bölümünde kaza analitikleri ile bir kaza tahmin modeli üzerinde çalışılmıştır. Çalışmanın son bölümünde ise trafik ağı, gerçeğe yakın biçimde ele alınarak ilgilenilen ağda optimal çevrimdışı rota belirlemek için önerilen çözüm yaklaşımı açıklanmış ve çalışmanın sonuçları tartışılmıştır.

1.1. Veri Analitikleri

Veri analitikleri veriyi anlamlı ve kullanılabilir bilgiye dönüştürür. Veri analitikleri, verilerdeki kalıpları ve eğilimleri ortaya çıkarmak, görselleştirmek, keşfetmek ve bağlantı kurmak için istatistiksel teknikler (merkezi eğilim ölçüleri, grafikler vb.), bilgi sistem yazılımları (veri madenciliği) ve yöneylem araştırması (doğrusal programlama) metodolojilerinin kullanımını içeren bir süreç olarak tanımlanabilir (Schniederjans, Schniederians ve Starkey, 2014).

Veri analitiği yaklaşımları genellikle dört başlık altında incelenmektedir.

1. Tanımlayıcı analitikler: Tanımlayıcı istatistiksel tekniklerin kullanıldığı bu veri analitiği yaklaşımı verideki “Ne oldu” sorusunun yanıtını verir. Grafikler, merkezi eğilim ölçüleri, olasılık dağılımları, frekans dağılımları vb. istatistiksel yöntemler kullanılarak gerçekleşen olaylara ait verilerin anlaşılır hale gelmesi sağlanır.
2. Kestirimsel analitikler: Kestirimsel analitikler, geçmiş olaylardan gelecek olayları tahmin etmek amacıyla ileri istatistiksel, veri madenciliği ve yöneylem araştırması tekniklerini bir arada kullanan bir analitik türüdür. Başka bir deyişle, kestirimsel analitikler “Ne olacak?” sorusunun yanıtını verir. Buradan elde edilecek sonuçlar, firmanın ürün ve hizmetlerini iyileştirmek için yararlanabileceği fırsatları tahmin etmektedir (Ajah ve Nweke, 2019).
3. Teşhis analitiği: Gerçekleşen olayların veya problemlerin nedenlerini ortaya çıkarmak için geçmiş verilerin analiz edilerek sorunun derinliklerine inilmesine yardımcı olur. Teşhis analitiğinin içeriğini oluşturan “Neden oldu?” sorusunun yanıtını bulmak üzere tanımlayıcı analitikleri kullanır.
4. Kuralcı analitikler: Bu analitikler, tahsis edilen kaynakları en iyi şekilde kullanmak için karar bilimi, yönetim bilimi ve yöneylem araştırması metodolojilerinin gücünü kullanır (Ajah ve Nweke, 2019). “Hangi süreç uygulanmalı?”, “Nasıl hareket edilmeli?” gibi bir durumda uygulanması gereken süreci adım adım belirleyen analitik türüdür. Veri seti üzerinden gelecek ile ilgili ne olacağının belirlenmesinin ardından gerçekleşebilecek olaylara karşı ne tür eylemler yapılabileceğine yönelik bilgiler sunar (Keskin ve Yıldız, 2018). Örneğin bir sürücü Google Haritalar’dan daha kolay bir rota aradığında kuralcı analitikler devreye girer (Acadgild, 2019).

Bu analitik yaklaşımlar, karar verme konusunda bilgi sağlamak için bağımsız olarak veya birleştirilmiş halde kullanılabilir (Ajah ve Nweke, 2019).

Önceki bölümde trafik ile ilgili bahsedilen birçok problemi çözmek için kurulan akıllı ulaşım sistem çözümleri tarafından toplanan veriler, veri analitiği ile anlamlandırılabilir. Örneğin, trafik sıkışıklıklarının gerçekleşmeden önce geçmiş verilerden hareketle tahmin edilmesi bir kestirimsel analitik uygulamasıdır. Benzer şekilde Narishman ve diğerleri (2017) geçmiş kaza verilerini kullanarak gelecekte kaç adet kaza olacağını tahmin etmek için bir kestirimsel analitik uygulaması önermişlerdir.

Ulaşım filolarının emisyonlarını en aza indirmek amacıyla Wu ve Yang (2017) kuralcı analitiklerin bir parçası olarak optimizasyon modeli önermişlerdir.

1.2. İçerik ve Ana Katkı

Bu çalışmada trafik bir bütün olarak ele alınmıştır. Özellikle Eskişehir kapsamında gerçekleştirilen çalışmada trafik uygulamaları için tanımlayıcı, kestirimsel ve kuralcı analitik uygulamaları yapılmıştır. Trafik kazalarının karakteristiklerini anlayabilmek için veri madenciliği ve temel istatistiksel teknikler kullanılarak tanımlayıcı analitikler oluşturulurken; trafik akışlarını tahmin etmek üzere istatistiksel ve makine öğrenmesi yöntemleri kullanılarak kestirimsel analitik çalışması uygulanmıştır. Çalışmanın son bölümünde üzerinde çalışılan trafik ağında rota bulma çalışması ise kuralcı analitikler kapsamına girmektedir. Burada önerilen kuralcı analitik yaklaşımı bir optimizasyon aracı içermektedir.

Trafik akış sayımları bir zaman serisi olarak ele alınmış ve 15 dakikalık zaman periyotları için bir Yapay Sinir Ağı (YSA) modeli önerilmiştir. Önerilen YSA modelinin mimarisi ve eğitim sürecinin optimizasyonu için bir metasezgisel ile melezlenmiştir. Seçilen metasezgisel daha önce literatürde kesikli ve sürekli optimizasyon problemlerinde Genetik Algoritmadan daha başarılı olduğu gösterilmiş olan Yanlı Rastgele Anahtarlı Genetik Algoritmalarıdır. YSA modelinin optimizasyonu ile ilgili literatürdeki çalışmalar incelendiğinde daha önce YSA ile Yanlı Rastgele Anahtarlı Genetik Algoritmaların melezlendiği bir başka çalışmaya rastlanılmamıştır.

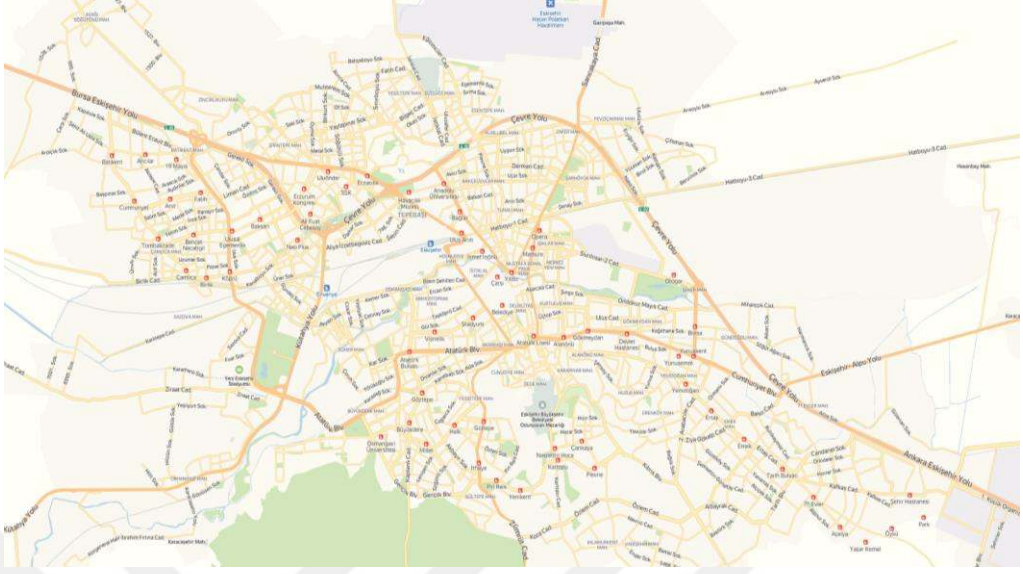
Eskişehir ilinde trafik güvenliği seviyesinin artırılması amacıyla trafik kazalarının karakteristiklerini ortaya çıkarmak için bazı tanımlayıcı istatistik ve veri madenciliği uygulamaları yapılmıştır. Gerçek hayattan alınan kaza kayıtları işlenebilir hale getirilmiş ve anımladırılmıştır. Kazalara ait karakteristikleri belirlemek için kümeleme analizi yapılmıştır. Buna ek olarak, trafik kazalarını tahmin etmek amacıyla makine öğrenmesi temelli sınıflandırma modelleri oluşturulmuştur. Trafik kaza tahminleme problemi literatürde daha önce benzerine rastlanmamış olan tek sınıf sınıflandırma problemi olarak ele alınmıştır. Trafik kaza verisetleri genellikle sadece kaza içeren durumları içerdiğinden diğer sınıflandırıcıları test etmek için bir ters örnek oluşturma prosedürü önerilmiştir. Son olarak, Eskişehir’de gerçekleşen trafik kazalarının gerçekleşmelerine ait olasılık dağılımları ortaya çıkarılmaya çalışılmıştır.

Çalışmanın son bölümünde, trafikte rota önerisi oluşturabilmek için ilk iki bölümde elde edilen çıktıların dahil edildiği bir Markov Karar Süreci temelli çözüm yöntemi önerilmiştir. Bu bölümde trafik, kavşaklar arası seyahat süresinin stokastik ve kavşakların dinamik olduğu varsayımları altında ele alınmıştır. Modelde iki kavşak arasındaki seyahat süresi, sabit hızda gidildiğinde gerçekleşen seyahat süresine ek olarak geçmişten elde edilen trafik akış yoğunluğu olasılıkları ve kaza gerçekleşme olasılıkları ile trafik ışığına yakalanma olasılığına bağlı olarak değişir. Trafik ağının dinamikliği ise ikinci ve üçüncü bölümlerde elde edilmiş trafik akış ve kaza tahminlerinin doğru olduğu varsayımı altında sağlanmıştır. Bir kavşakta trafik akışının yoğun olduğu ya da bir yolda ölümlü ve yaralanmalı kaza gerçekleştiği tahmin edildiğinde; kavşağın ilgili zaman dilimi için trafik ağından çıkarılması ve tersi durumda ise kavşağın ağı geri eklenmesi ile trafik ağı dinamik bir ağ olarak düşünülmüştür. Ek olarak, problemin amaç fonksiyonuna eklenen trafik ışığına yakalanma durumu ile gerçek trafik ağı temsil edilmeye çalışılmıştır. Literatürde bu konuda yer alan çalışmalar incelendiğinde çevrimdışı rotalama söz konusu olduğunda özellikle kaza ile ilgili tahminlerin kullanılmadığı görülmüştür. Yapılan çalışmalar çoğunlukla çevrimiçi durumlarda sürekli anlık bilgi alınan ya da rotalama tamamen çevrimdışı ise stokastik seyahat süresi içeren yaklaşımlar olarak düşünülmüştür. Çalışmanın bu konudaki katkısı çevrimdışı rotalamada trafiği engelleyebilecek daha fazla durumu çözüm yaklaşımına yansıtmaktır.

Sonuç olarak, bu çalışmada bir trafik ağının doğal akışını engelleyebilecek durumlardan bazıları göz önüne alınarak bir rota önerisi oluşturabilmek için bütünsel bir çözüm yaklaşımı uygulanmıştır.

1.3. Ele Alınan Yol Ağı

Bu çalışmada uygulama yeri olarak Eskişehir seçilmiştir. Eskişehir'in seçilme sebebi, şehre kurulmuş olan akıllı kavşak sistemi ile bu sistemi yöneten bir trafik kontrol merkezinin bulunmasıdır. Eskişehir şehir içi trafik ağı bulvar, cadde ve sokak toplamı olarak 3575.782 km'dir (Eskişehir Büyükşehir Belediyesi, 2016). Eskişehir yol ağı Şekil 1.1'de verilmiştir.



Şekil 1.1. Eskişehir yol ağı

Eskişehir Büyükşehir Belediyesi'ne bağlı olarak çalışan akıllı trafik kontrol merkezi, Eskişehir bünyesinde başlangıçta 17 adet akıllı hale getirilmiş kavşak vasıtasıyla trafiği incelemektedir. Akıllı kavşak olarak adlandırılan kavşaklara yerleştirilmiş olan balık gözü kameralar aracılığıyla kavşaklara ait tüm yönlerdeki araç sayıları her 15 dakikada bir toplanmaktadır. Şekil 1.2'de bir kavşağa ait kamera görüntüsü yer almaktadır.



Şekil 1.2. Bir kavşağa ait kamera görüntüsü

Kameralar aracılığıyla kavşaklara ait tüm yönler izlenmekte ve sayımlar yön bazında elde edilmektedir. Trafik akış sayımları araç bazında yapılmakta, bisikletten

tramvaya kadar araç türlerine ait sayım bilgisini içermektedir. Merkez, trafik akış sayımlarını toplayarak kavşakları incelemek ve yoğun kavşaklarda bekleme süresini azaltmak amacıyla çalışmalar yapmaktadır.

Akıllı Trafik Kontrol Merkezi aynı zamanda trafik kazaları ile ilgili bilgileri de barındırmaktadır. Trafik kaza bilgileri kaza esnasında tutulan kaza raporları aracılığıyla elde edilmektedir. Eskişehir trafik güvenliği açısından incelendiğinde Emniyet Genel Müdürlüğünün yayınlamış olduğu Trafik İstatistik Bülteni'nde (Emniyet Genel Müdürlüğü, 2018) belirtildiği üzere Eskişehir, yaşanan trafik kaza sayılarında 81 il içinde 23. sırada yer almaktadır. Bu durum da Eskişehir'in trafik güvenliği seviyesinin düşük olduğunu göstermektedir. Ayrıca 2018'de Eskişehir'de gerçekleşen ölümlü ve yaralanmalı kaza sayısı 2298 iken, bu kazalarda 68 ölüm ve 3680 yaralanma gerçekleşmiştir. Ek olarak, 2744 adet maddi hasarlı kaza gerçekleşmiştir. Bu ağır bilanço sebebiyle Eskişehir'de yaşanan trafik kazaları incelenerek sebeplerinin veya karakteristiklerinin belirlenmesi gerekmektedir.

Sonuç olarak, bu çalışmada trafik kontrol merkezinden temin edilen bilgiler aracılığıyla trafik sorunlarına çözüm bulmak amacıyla Eskişehir için daha önce bahsedilmiş veri analitiği çalışmaları gerçekleştirilmiştir.

2. TRAFİK AKIŞ TAHMİNLEMESİ

Trafik akışı, yolların ya da kavşakların belirlenmiş noktalarından belirli zaman aralıklarında geçen araç sayısı olarak ifade edilebilir. Trafik akışı sayımları sokak kameraları, küresel konumlama sistemi (GPS), sensörler vb. araçla toplanabilir. Veriler genellikle 15 dakika gibi belirli zaman periyotlarında toplanan veriyi içerir.

Trafik akışı genellikle periyodiktir ve döngü genellikle bir gündür. Çoğunlukla, trafik yoğun saatleri sabah ve öğleden sonra yaşanır. Farklı günlerdeki trafik akışı değişkenliği genellikle benzerdir. Sonuç olarak, mevsimsellik, trafik akışı için dikkate alınabilecek bir faktördür (Yu ve Li, 2017). Şekil 2.1’de belirli bir noktaya ait 15 dakikada bir toplanan 2 günlük trafik akışında da bahsedilen özellikler gözlenmektedir. Trafik akışı tüm bu özellikleriyle zaman serisi olarak ele alınır.



Şekil 2.1. İki günlük trafik akışı

2.1. Zaman Serileri

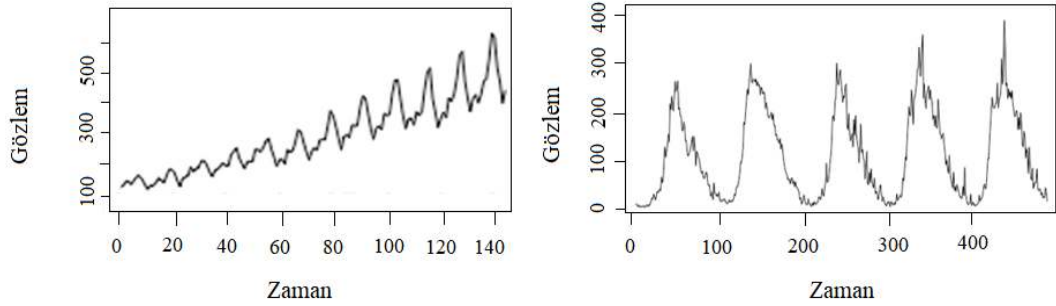
Bir zaman serisi, her biri belirli bir zamanda kaydedilmiş olan bir dizi gözlemdir (Brockwell ve Davis, 2016). Gözlemler sabit zaman aralıkları için kaydedilir. Zaman serileri çoğunlukla kesikli olarak ele alınır. Bir ürüne ait aylık satış miktarları, bir kavşağa ait trafik akış sayımları, bir şehrin günlük hava sıcaklık değerleri, bir finansal göstergenin 1 dakikalık değerleri, bir şehre ait günlük elektrik tüketimi miktarı gibi aynı periyotlarla aynı değişken için toplanan veriler zaman serilerine örnektir.

Zaman serileri trend, mevsimsellik, döngü ve hata olmak üzere dört tür bileşenden oluşabilir. Bir zaman serisi bu bileşenlerin bazılarını ya da tümünü içerebilir. Uzun dönemli bir bileşen olan trend, zaman serisinde görülen pozitif veya negatif eğilimi ifade

eder. Mevsimsel bileşen, serinin aynı zaman periyodlarında (aynı ay, gün, vb.) düzenli dalgalanmalar içermesi durumunda söz konusudur. Veride görülen sabit frekansta olmayan birbirini takip eden artış ve azalışlar ise döngü bileşenini oluşturur. Zaman serisi tüm bu bileşenlerden arındırıldığında, kalan son kısım ise hata olarak adlandırılır. Bir zaman serisi y_t , Denklem (2.1)'de verildiği şekilde ifade edilebilir.

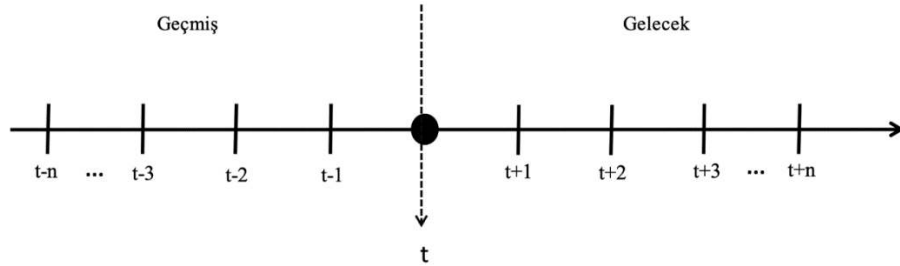
$$y_t = T_t + S_t + C_t + \varepsilon_t \quad (2.1)$$

Burada T_t, S_t, C_t ve ε_t sırasıyla trend, mevsimsellik, döngü ve hata kısımlarını ifade etmektedir. Şekil 2.2'de zaman serileri bileşenleri için örnekler verilmiştir. Şekil 2.2'nin birinci kısmında mevsimsellik ve trend, ikinci kısmında ise döngü bileşenleri açıkça görülmektedir.



Şekil 2.2. Mevsimsellik, trend ve döngü içeren zaman serileri için örnekler

Zaman serisi verilerini tahmin ederken amaç, gözlem dizisinin geleceğe nasıl devam edeceğini tahmin etmektir (Hyndman ve Athanasopoulos, 2018). Bir zaman serisinde t anında gelecek için yapılacak tahminler, geçmiş veriler aracılığıyla elde edilir. Şekil 2.3'te bir zaman serisine ait zaman çizelgesini göstermektedir.



Şekil 2.3. Zaman serisi çizelgesi

2.2. Zaman Serileri Tahmin Yöntemleri

Zaman serilerinde tahminler geçmiş döneme ait veriler aracılığıyla yapılır. Geçmiş verilerin sınırlı ya da yeni ürünlerin satışa sunulmasındaki gibi geçmiş veri olmadığı durumlarda yargısal tahminler yapılabilir. Delphi metodu, ilgili konuda uzman kişilerin bireysel olarak yapmış olduğu tahminlerden birleştirilerek nihai tahminin elde edildiği yargısal tahmin yöntemlerinden biridir. Zaman serileri tahmininde kullanılan yöntemler geleneksel yöntemler ile makine öğrenmesi başlıkları altında incelenmiştir.

2.2.1. Geleneksel yöntemler

Zaman serisi tahminlemesinde kullanılan geleneksel yöntemler hareketli ortalamalar, üstel düzeltme, Holt-Winter üstel düzeltme yöntemi ve regresyondur. Otoregresif modeller de bu kategoriye dahil edilebilir.

Hareketli ortalamalar, zaman serileri tahmininde ve ayrıştırmasında kullanılan genel bir yöntemdir. Hareketli ortalamalar yönteminde gelecek dönem tahmini, geçmiş n dönemin ortalaması alınarak elde edilir. $\hat{y}_{t+1}$, gelecek dönemin tahmin değeri n adet geçmiş veri $(y_t, y_{t-1}, \dots, y_{t-n})$ ile Denklem (2.2)'de belirtildiği şekilde elde edilir.

$$\hat{y}_{t+1} = \frac{1}{n} \sum_{i=1}^n y_{t-i} \quad (2.2)$$

Üstel düzeltme yöntemi 1950'li yıllarda Holt (Holt, 2004) tarafından önerilmiştir. Üstel düzeltme yöntemleri kullanılarak üretilen tahminler, geçmiş gözlemlerin ağırlıklı ortalamalarıdır ve ağırlıklar, gözlemler geçmişe gittikçe üstel olarak azalır. Başka bir deyişle, gözlem ne kadar tahmin edilen zamana yakınsa, gözleme ilişkin ağırlık da o kadar yüksek olur (Hyndman ve Athanasopoulos, 2018). Bir t anında 1 dönem ilerisi için yapılacak tahmin $\hat{y}_{t+1}$ üstel düzeltme yöntemi ile Denklem (2.3)'teki şekilde hesaplanır:

$$\hat{y}_{t+1} = \alpha y_t + (1 - \alpha) y_{t-1} \quad (2.3)$$

Denklem (2.3)'te yer alan α , $[0,1]$ aralığında değer alan üstel düzeltme parametresidir.

Holt ve Winter, üstel düzeltme yöntemini mevsimselliği de içerecek şekilde genişleterek Holt-Winter yöntemini sunmuşlardır (Winters, 1960). Mevsimsellik içeren toplamsal bir zaman serisi için Denklem (2.4), (2.5), (2.6) ve (2.7) geliştirilmiştir.

$$y_{t+h|t} = l_t + hb_t + s_{t+h-m(k+1)} \quad (2.4)$$

$$l_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)(l_{t-1} + b_{t-1}) \quad (2.5)$$

$$b_{t-1} = \beta^*(l_t - l_{t-1}) + (1 - \beta^*)b_{t-1} \quad (2.6)$$

$$s_t = \gamma(y_t - l_{t-1} - b_{t-1}) + (1 - \gamma)s_{t-m} \quad (2.7)$$

Denklem (2.4)'teki k , tahmin için kullanılan mevsimsel endekslerin tahminlerinin serinin son yılından gelmesini sağlayan $(h - 1/m)$ 'in tamsayı kısmıdır. Seviye denklemi, t anı için mevsimsellikten arındırılmış gözlem olan $(y_t - s_{t-m})$ ile mevsimsel olmayan tahmini ifade eden $(l_{t-1} + b_{t-1})$ 'in ağırlıklı ortalamasıdır. Trend denklemi Holt'un doğrusal yöntemiyle aynıdır. Mevsimsel denklem ise, t anının mevsimsel indeksi olan $(y_t - l_{t-1} - b_{t-1})$ ile bir önceki dönemin aynı mevsiminin (örneğin m periyod öncesi) ağırlıklı ortalamasıdır (Hyndman ve Athanasopoulos, 2018).

Zaman serilerini tahmin etmenin bir başka yolu da doğrusal ya da doğrusal olmayan regresyondur. Doğrusal regresyon ile bir y zaman serisini tahmin etmek için diğer zaman serileri ile doğrusal bir ilişkisi olduğunu varsayımı yapılır. Örneğin, bir günlük sıcaklık değeri ve haftanın günü verileri kullanılarak günlük elektrik talebi regresyon yöntemi ile tahmin edilebilir. En basit durumda, regresyon modeli tahmin edilen bağımlı y değişkeni ile bağımsız x değişkeni arasında incelenen aralıktaki her t dönemi için Denklem (2.8)'deki doğrusal ilişkiyi kurar:

$$y_t = \beta_0 + \beta_1 x_t + \varepsilon_t \quad (2.8)$$

β_0 , $x_t = 0$ iken y 'nin tahmin değerini verirken; β_1 , x 'teki 1 birim artışın y 'nin ortalama tahmin değerindeki değişikliği ifade eder. Rastgele hata olarak adlandırılan ε_t ise doğrusal modelden sapmayı gösterir (Hyndman ve Athanasopoulos, 2018).

Geleneksel tahmin yöntemlerinden biri olan otoregresif birleştirilmiş hareketli ortalamalar yöntemi çalışmada ele alındığından Bölüm 2.1.1.1'de detaylı olarak açıklanmıştır.

2.2.1.1. Otoregresif birleştirilmiş hareketli ortalamalar yöntemi

Otoregresif Birleştirilmiş Hareketli Ortalamalar Yöntemi (ARİMA), ilk kez Box ve Jenkins tarafından önerilmiştir. ARİMA süreçleri, zaman serilerini analiz etmek için kullanılan bir tür stokastik süreçtir (Conteras vd., 2003). Bahsedilen stokastik süreçler

otoregresif (AR), hareketli ortalamalar (MA), otoregresif hareketli ortalamalar (ARMA) ve otoregresif birleştirilmiş hareketli ortalamalar (ARİMA) yöntemleri ile ele alınabilir.

ARİMA modellerinde durağan olmayan bir zaman serisi, veri noktalarının sonlu farkları uygulanarak durağan hale getirilmektedir (Adhikari ve Agrawal, 2013). Bir ARİMA modelinde bir değişkenin gelecekteki değeri, geçmiş gözlemlerin ve rastgele hataların doğrusal bir fonksiyonu olarak ifade edilir. Başka bir deyişle bir zaman serisinin altında yatan yapı Denklem (2.9)'deki gibi ifade edilebilir:

$$y_t = \theta_0 + \varphi_1 y_{t-1} + \varphi_2 y_{t-2} + \dots + \varphi_p y_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \dots - \theta_q \varepsilon_{t-q} \quad (2.9)$$

Burada y_t , ε_t sırasıyla t zaman periyodundaki gerçek değeri ve rastgele hatayı; φ_1 ($i=1,2,\dots,p$) ve θ_j ($j=1,2,\dots,q$) ise model parametrelerini ifade eder (Zhang, 2003). Bir ARİMA modelinin d parametresi ise durağanlık için ihtiyaç duyulan mevsimsel olmayan farkların sayısıdır. d parametresine göre yapılan fark alma işlemi ile zaman serisi mevsimsellik bileşeninden arındırılır.

ARİMA'nın parametreleri, p , d ve q sıfıra eşit veya daha büyük tam sayılar olduğundan ve modelin otoregresif, tümleşik ve hareketli ortalama kısımlarını ifade eder.

ARİMA modeli, mevsimsel faktörler bulunduğunda mevsimsel ARİMA (SARİMA) modeline dönüşür. Mevsimsel otoregresif notasyonu P , mevsimsel hareketli ortalama notasyonu Q olmak üzere bir SARİMA süreci $(p,d,q)(P,D,Q)s$ parametreleri ile ifade edilir. Burada s mevsimsel periyodun uzunluğudur (Permanasari, Hidayah, & Bustoni, 2013). B 'nin backshift operatörünü ifade ettiği SARİMA modelinin genel gösterimi Denklem (2.10)'da verilmiştir.

$$\varphi_p(B)\varphi_p(B^s)y_t = \theta_0 + \theta_q(B)\theta_q(B^s)\varepsilon_t \quad (2.10)$$

2.2.2. Makine öğrenmesi yöntemleri

Zaman serilerini tahmin etmek için makine öğrenme algoritmaları da kullanılabilir. Yapay Sinir Ağları ve türevleri, K-en yakın komşu regresyonu, destek vektör regresyon, CART regresyon ağaçları ve Gaussian süreçleri ile zaman serileri tahminleri yapılabilir (Makridakis, Spiliotis ve Assimakopoulos, 2018). Bu çalışmada makine öğrenmesi yöntemlerinden Yapay Sinir Ağları ve Destek Vektör Regresyon kullanıldığından bu yöntemler detaylı açıklanacaktır.

2.2.2.1. Destek vektör regresyon

Destek Vektör Regresyon (DVR), sınıflandırma için makine öğrenmesi algoritmalarından biri olan Destek Vektör Makinelerinin (DVM) bir türevidir. DVM, Vapnik tarafından sınıflandırma için sunulmuştur ve Schölkopf ve diğerleri (Schölkopf vd., 2001) tarafından regresyon modellemesi için genişletilmiştir. Belirli bir eğitim seti olan $(x_1, y_1), \dots, (x_t, y_t)$ ifadesinde y_i , x_i 'nin hedef değeridir. DVR, bu örnek için bir regresyon modeli eğitir ve modeli gelecek hedef verilerini tahmin etmek için kullanır. DVR fonksiyonu Denklem (2.11)'deki gibi formüle edilmiştir.

$$f(x) = w^T \varphi(x) + b \quad (2.11)$$

Burada x , girdi uzayından öznitelik uzayına doğrusal olmayan bir eşleştirmedir. W ise ağırlık katsayılarının vektörü ve b sapma sabitidir. W ve b Denklem (2.12), (2.13) ve (2.14)'te verilen optimizasyon probleminin enküçüklenmesi ile elde edilir:

$$\text{minimum } \frac{1}{2} \|w\|^2 \quad (2.12)$$

$$y_i - (\langle w, \varphi(x_i) \rangle + b) \leq \varepsilon \quad (2.13)$$

$$(\langle w, \varphi(x_i) \rangle + b) - y_i \geq \varepsilon \quad (2.14)$$

Uygunluk sorunları ile başa çıkmak ve yöntemi daha güçlü hale getirmek için duyarlı noktalar çıkartılmaz. Yerine bu noktalar Denklem (2.15), (2.16), (2.17) ve (2.18)'de verildiği gibi ξ_i, ξ_i^* aylak değişkenleri ile cezalandırılır.

$$\text{minimum } \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l (\xi_i + \xi_i^*) \quad (2.15)$$

$$y_i - (\langle w, \varphi(x_i) \rangle + b) \leq \varepsilon + \xi_i \quad (2.16)$$

$$(\langle w, \varphi(x_i) \rangle + b) - y_i \geq \varepsilon + \xi_i^* \quad (2.17)$$

$$\xi_i, \xi_i^* \geq 0 \quad (2.18)$$

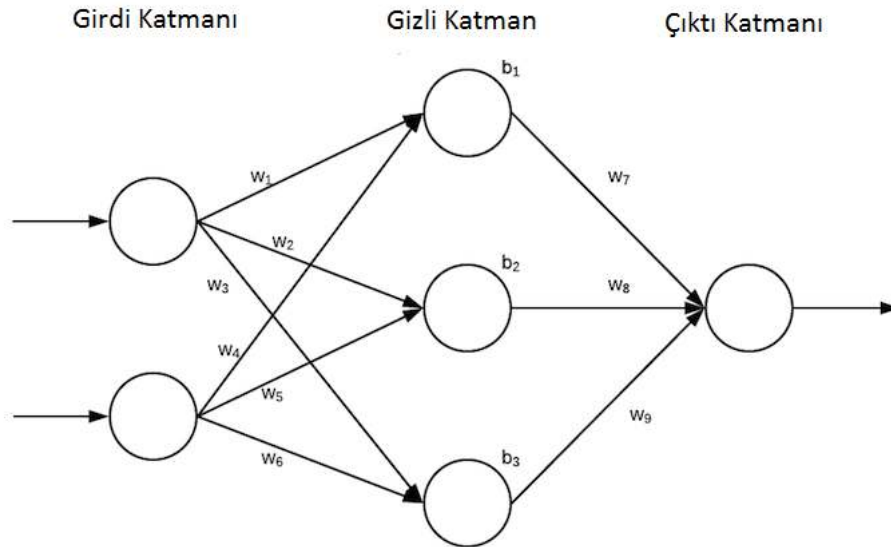
Burada $C > 0$ maliyet sabiti, model karmaşıklığı ile eğitim hatası arasındaki dengeyi belirler.

$$f(x) = \sum_{i=1}^l (\alpha_i + \alpha_i^*) K(x_i, x) + b \quad (2.19)$$

Denklem (2.19), problemin dualinin çözümüdür. α_i, α_i^* sınıf olmayan Lagrange çarpanlarıdır ve $K(x_i, x)$ iç çarpımı ifade eden kernel fonksiyonudur. Kernel fonksiyonları, iç çarpımın yüksek boyutlu bir öznitelik uzayından aktarılmasını sağlar. Gelen olarak kullanılan kernel fonksiyonları doğrusal, polinom, Gaussian fonksiyonları ile radyal tabanlı fonksiyondur (RBF).

2.2.2.2. Yapay sinir ağları

Biyolojik sistemlerden ve özellikle insan beyniyle ilgili araştırmalardan esinlenen YSA deneyimlerden öğrenebilir ve genelleştirebilir (Zhang, Patuwo, ve Hu, 1998). Standart bir YSA modeli bir girdi, bir gizli ve bir çıktı katmanından oluşur. Şekil 2.4'te bir girdi katmanı, bir gizli katman ve bir çıktı katmanı içeren bir YSA modeli örneği verilmiştir.



Şekil 2.4. Üç katmanlı bir YSA örneği

Verilen YSA modelinde w_i ($i=1, \dots, 9$) nöronlar arasındaki bağlantı ağırlıklarını, b_j ($j=1, 2, 3$) ise eşik değerin gizli nöronlara ait ağırlık değeridir. YSA modelinin ağırlıklarının belirlenmesi süreci eğitim olarak adlandırılır. YSA'lar denetimli öğrenme mekanizmalarıdır. Başka bir deyişle, ağ bir eğitim aşamasında verilen örneklerin çıktıları ile eğitilir ve eğitim sonucu elde edilen model kullanılır. YSA'ların eğitimi sonucunda bağlantılar arasındaki ağırlıklar ve eşik değerleri ağırlıkları belirlenmiş olur.

YSA'ların eğitimi amaç fonksiyonu minimum hata olan bir optimizasyon problemidir. Hata fonksiyonu olarak hedeften sapmanın karelerinin ortalaması olarak hesaplanır. Bu doğrusal olmayan problemi çözmek için genellikle geriyayılım algoritması kullanılır. Geriyayılım algoritması, YSA'ların eğitiminde en yaygın olarak kullanılan algoritmadır. Geriyayılım algoritması, gradyan iniş yöntemini kullanarak hata fonksiyonunun minimumunu arar. Hata fonksiyonunu minimize eden ağırlık kombinasyonu öğrenme probleminin çözümü olarak kabul edilir (Rojas, 1996).

YSA'larda ileriye ve geriye doğru olmak üzere iki tür hesaplama yapılır. İleriye doğru hesaplama ile YSA'nın çıktı değeri hesaplanırken; geriye doğru hesaplama aracılığıyla elde edilen çıktının gerçek değerden farkı olan hata değerine göre nöronlar arasındaki bağlantı ağırlıklarının güncellenmesi sağlanır.

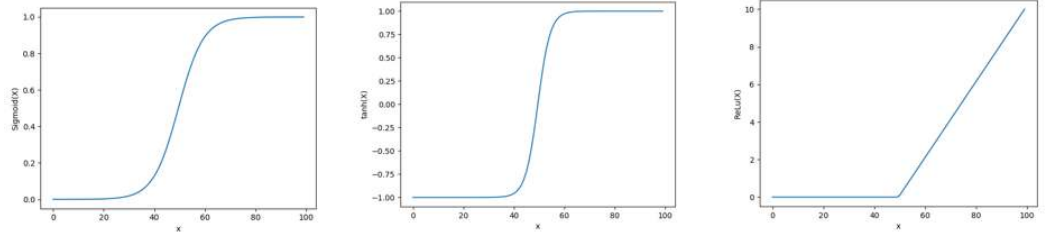
Şekil 2.4'te görülen üç katmanlı YSA örneği üzerinden YSA'ların çalışma prensibi anlatılacaktır. İleri hesaplamasının ilk aşamasında her bir gizli nöronun değerini hesaplamak için girdi değerleri ile ilgili nörona ait ağırlıklar çarpılarak her bir i gizli nöronu için h_i ile gösterilmiş toplam değerler Denklem (2.20a), (2.20b), (2.20c) ile hesaplanmaktadır.

$$h_1 = i_1 \times w_1 + i_2 \times w_4 + (1 * b_1) \quad (2.20a)$$

$$h_2 = i_1 \times w_2 + i_2 \times w_5 + (1 * b_2) \quad (2.20b)$$

$$h_3 = i_1 \times w_3 + i_2 \times w_6 + (1 * b_3) \quad (2.20c)$$

Nöronlara ait toplam değerler hesaplandıktan sonra her bir nöron için aktivasyon fonksiyonu uygulanır. Aktivasyon fonksiyonu olarak türevlenebilir doğrusal olmayan bir fonksiyon kullanılır. Doğrusal olmayan aktivasyon fonksiyonu YSA'nın girdi ile çıktısı arasında doğrusal olmayan bir ilişki kurar, aksi halde sinir ağı bir doğrusal regresyon modeli gibi çalışır. En çok kullanılan aktivasyon fonksiyonlardan bazıları sigmoid, hiperbolik tanjant ve Rektifiye Lineer Birimler (ReLU) fonksiyonlarıdır. Şekil 2.5'te bahsedilen fonksiyonlara ait fonksiyon grafikleri verilmiştir.



Şekil 2.5. Sigmoid, tanh ve ReLU aktivasyon fonksiyonları

Her bir gizli nöron için aktivasyon işlemi sigmoid fonksiyonu kullanılarak Denklem (2.21a, 2.21b, 2.21c)'deki gibi hesaplanır.

$$h_1 = \frac{1}{1 + e^{-h_1}} \quad (2.21a)$$

$$h_2 = \frac{1}{1 + e^{-h_2}} \quad (2.21b)$$

$$h_3 = \frac{1}{1 + e^{-h_3}} \quad (2.21c)$$

İleri hesaplamamanın son aşamasında, gizli nöronlarda aktivasyon uyguladıktan sonra elde edilen gizli nöron değerleri ilgili ağırlıklarla çarpılır ve elde edilen değerler toplanarak çıktı değeri o Denklem (2.22)'deki verildiği şekilde elde edilir.

$$o = w_7 \times h_1 + w_8 \times h_2 + w_9 \times h_3 \quad (2.22)$$

YSA modelinin çıktısı elde edildikten sonra hedef değere göre ne kadar hata yapıldığı hesaplanmalıdır. Çıktı sayısı birden fazla ise her bir çıktı için elde edilen hata değerleri toplanır. Hata fonksiyonu olarak genellikle hataların kareleri ortalaması (MSE) kullanılır. Toplam hata değeri E ile ifade edilmektedir.

Hata değeri hesaplandıktan sonra geriyealım algoritması ile bağlantı ağırlıkları güncellenir. Bunun için geriye doğru hesaplama yapılır. Öncelikle hatanın her bir ağırlığa göre türevi alınır. Bu işlem Denklem (2.23a), (2.23b) ve (2.23c)'de verilen “zincir kuralı” adı verilen bir kural ile yapılır.

$$\frac{\partial E}{\partial w_7} = \frac{\partial E}{\partial o} * \frac{\partial o}{\partial h_1} * \frac{\partial h_1}{\partial w_7} \quad (2.23a)$$

$$\frac{\partial E}{\partial w_8} = \frac{\partial E}{\partial o} * \frac{\partial o}{\partial h_2} * \frac{\partial h_2}{\partial w_8} \quad (2.23b)$$

$$\frac{\partial E}{\partial w_9} = \frac{\partial E}{\partial o} * \frac{\partial o}{\partial h_3} * \frac{\partial h_3}{\partial w_9} \quad (2.23c)$$

Hesaplanan türevler aracılığıyla ağırlık değerleri Denklem (2.24a), (2.24b) ve (2.24c)'de verildiği şekilde güncellenir.

$$w_7 = w_7 - \eta * \frac{\partial E}{\partial w_7} \quad (2.24a)$$

$$w_8 = w_8 - \eta * \frac{\partial E}{\partial w_8} \quad (2.24b)$$

$$w_9 = w_9 - \eta * \frac{\partial E}{\partial w_9} \quad (2.24c)$$

η , öğrenme katsayısıdır. Öğrenme katsayısı sabit olarak verilebileceği gibi adaptif olarak da belirlenebilir. Benzer şekilde girdi nöronları ile gizli nöronlar arasındaki ağırlıklar da güncellenir ve böylece YSA eğitiminin bir iterasyonu tamamlanmış olur.

2.3. Literatür Taraması

Trafik akışı tahminlemesi üzerine literatürde sunulan çalışmaların çoğu istatistiksel, metasezgisel ve makine öğrenme ile oluşturulmuş melez yöntemleri içermektedir. Bu konuda geliştirilen melez yöntemlerde YSA kullanımı oldukça yaygındır.

Chan ve diğerleri (2011), üstel düzeltme, YSA ve GA'ya dayanan bir trafik akış tahmin modeli kurmuşlardır. Önerdikleri modelde üstel düzeltme yöntemi ile veriye ön işleme uyguladıktan sonra YSA ile tahmin aşamasında modeli parametrelerini GA ile optimize etmişlerdir. Shen, Kong ve Chen (2011), kısa dönemli trafik akış tahmini için Kalman Filtresi ve YSA ile ayrı ayrı oluşturdukları tahminleri birleştirdikleri bir bulanık birleştirme modeli yaklaşımı sunmuşlardır. Moretti ve diğerleri (2015), kentsel trafik akışını tahmin etmek üzere melezledikleri topluluk yaklaşımli YSA modeli ile haftalık akış dağılımı temelli bir istatistiksel yöntem ile oldukça başarılı sonuçlar elde etmişlerdir. Chan ve diğerleri (2013) ise kısa dönemli trafik akış tahmini için Parçacık Sürü Optimizasyonu (PSO) ve YSA'yı entegre eden bir model sunmuşlardır.

Trafik akış tahminlemesi için önerilen melez yöntemlerin bir kısmı da DVR temellidir. Hu ve diğerleri (2016), trafik akış tahmini için oluşturdukları DVR modelinin

parametrelerini PSO ile optimize etmeye çalışmışlar ve önerdikleri algoritmanın DVR, YSA ve ARİMA yöntemlerinden daha iyi performans gösterdiğini öne sürmüşlerdir. Benzer şekilde Cheng ve diğerleri (2017) yine trafik akış tahminleme problemi için DVR'yi parametrelerini optimize etmek amacıyla altuzay ortogonal güvercin algoritması ile melezlemişler; önerdikleri algoritma ile başarılı tahminler elde edebilmişlerdir. Cai ve diğerleri (2019) ise geliştirdikleri DVR temelli trafik akış tahmini modeli için DVR parametrelerini yerçekimi arama algoritması ile belirlemişlerdir.

Literatürde trafik akış tahminlemesi için farklı yaklaşımlar da yer almaktadır. Habtemichaela ve Cetin (2016) gelişmiş bir K -en yakın komşu algoritması kullanılarak benzer trafik modellerini tanımlamaya dayanan kısa vadeli trafik tahmini için parametrik olmayan ve veriye dayalı bir yöntem önermişlerdir. Wang ve Ye (2015) ise trafik akışı zaman serisindeki veriler arasındaki değişen rastgele hatanın varyansın üstesinden gelebilmek için doğrusal ARİMA modeli ile doğrusal olmayan Açıklayıcı Genelleştirilmiş Otoresif Koşullu Değişen Varyans yöntemini birleştiren bir melez bir yöntem geliştirmişlerdir. Kumar ve Vanajakshi (2015), trafik akışı tahmini için mevsimsel ARİMA modeli önermişlerdir.

Literatürdeki çalışmalardan da görüldüğü üzere trafik akış tahmininde genellikle melez algoritmalar kullanılmıştır. Ayrıca, YSA'nın belirtilen tahmin probleminde sıklıkla kullanıldığı ve melezleri ile başarılı sonuçlar elde edildiği görülmüştür. Bu çalışmada da trafik akış tahminlemesi için YSA temelli bir algoritma üzerine çalışılmıştır.

2.4. Trafik Akış Tahminlemesi İçin Melez Bir Algoritma Önerisi

Önerilen melez algoritmanın amacı YSA mimarisini ve eğitimini optimize etmektir. YSA optimizasyonu adı altında uygun bir ağ yapısı bulmak ve optimal ağırlık değerlerini bulmak YSA'nın tasarımını zor optimizasyon problemleri haline getirir. Başka bir deyişle, YSA'ların başarısı büyük ölçüde mimariye, eğitim algoritmasına ve eğitimde kullanılan özelliklerin seçimine bağlıdır (Karaboga, Akay ve Ozturk, 2007). Yapay sinir ağlarının optimizasyonun iki ana konu ile karşılaşmaktadır. Bunlardan biri öğrenme optimizasyonu adı verilen denetimli öğrenme sürecinde nöronlar arasındaki bağlantı ağırlıklarının optimizasyonu iken; diğeri ağ yapısının optimizasyonu olarak adlandırılan ağın boyutunun veya öğrenme parametresinin optimizasyonudur (Vlahogianni, Karlaftis, ve Golias, 2005). Bu noktada yapılan çalışmalarda bu parametreler tek tek veya eşzamanlı olarak ele alınabilmektedir.

YSA'ların eğitim aşamasında geri yayılım gibi gradyan inişi tabanlı öğrenme algoritmaları kullanılır. Ancak, YSA'ların eğitimi için bu en eski yaklaşımın kullanılması, yerel minimuma takılmayı kolaylaştıran, düşük hassasiyetli ve yavaş yakınsama hızını içeren bazı dezavantajlara yol açmaktadır (Alwaisi ve Baykan, 2017). Bu dezavantajlar, YSA'ların öğrenme sürecinde farklı yöntemlerin araştırılmasına sebep olmuştur. YSA öğrenme süreci, amacı gerçek verilerden sapmayı ifade eden hatanın enküçüklenmesi olan bir optimizasyon problemi olarak ele alınabilir. Problem YSA'nın doğası gereği doğrusal değildir. Problemin doğrusal olmayışı, karar değişkenlerinin ve kısıtların fazlalığı sebepleriyle modelin kabul edilebilir sürelerde optimal değerini bulmak imkansızdır. Bahsedilen bu sebeplerden dolayı YSA'ların eğitim sürecinde kullanılan klasik öğrenme algoritmalarına alternatif olarak metasezgiseller uygulanabilir.

Literatürdeki pek çok çalışmada bu konuda metasezgisel algoritmalar sıklıkla kullanılmaktadır. Metasezgiseller, ağırlıklar, yapı, düğümler vb. YSA bileşenlerini bir optimizasyon problemi olarak formüle ederler (Ojha, Abraham ve Snasel, 2017). Literatürde YSA optimizasyonu konusunda yapılan çalışmalardan bazıları kullanılan yöntem ve optimize edilen YSA parametreleri ile Tablo 2.1'de özetlenmiştir.

Tablo 2.1. YSA optimizasyonu ile ilgili çalışmalar

Yıl	Yazarlar	Optimize edilen parametreler	Kullanılan metasezgisel	Ele alınan problem
2005	Vlahogianni, Karlaftis ve Golias (2005)	Adım sayısı, momentum katsayısı, öğrenme oranı, gizli nöron sayısı	Genetik Algoritmalar (GA)	Tahminleme
2007	Karaboga, Akay ve Ozturk (2007)	Bağlantı ağırlıkları	Yapay Arı Algoritması	XOR problemi
2011	Irani ve Nasimi (2011)	Bağlantı ağırlıkları ve eşik değeri	GA	Tahminleme
2013	Donate ve diğerleri (2013)	Girdi nöronu sayısı, gizli nöron sayısı, öğrenme katsayısı, ağırlıkların başlangıç değeri	GA, Diferansiyel Evrim Algoritması	Tahminleme
2013	Panigrahi, Karali ve Behera (2013)	Bağlantı ağırlıkları	GA, Diferansiyel Evrim Algoritması	Tahminleme
2014	Nur, Radzi ve Ibrahim (2014)	Bağlantı ağırlıkları	Harmoni Arama Algoritması	
2015	Chen, Do, Hsieh (2015)	Bağlantı ağırlıkları, eşik değeri	PSO, Guguk kuşu algoritması	Sınıflandırma
2017	Alwaisi ve Baykan (2017)	Bağlantı ağırlıkları	Shuffled frog leap algoritması	Sınıflandırma
2019	Moayed ve diğerleri (2019)	Bağlantı ağırlıkları	PSO	Tahminleme

Çalışmada YSA, Genetik Algoritmanın (GA) bir türü ile melezleneceğinden GA'nın kullanıldığı çalışmalara ağırlık verilmiştir. GA'nın YSA optimizasyonunda kullanıldığı çalışmaları özetleyen Chiroma ve diğerleri (2017), GA ile çoğunlukla başarılı sonuçlar elde edildiğini belirtmişlerdir. Islam ve diğerleri (2014), yük tahmininde kullanılmak üzere GA ile hibridlenmiş bir YSA önermişler ve çalışmalarında kromozomu

girdi ve gizli nöron sayısı ile bağlantı ağırlıkları için tanımlamışlardır. Irani ve Nasimi (2011), tahmin probleminde başlangıç ağırlık ve eşik değerlerini optimize etmek için önerdikleri hibrid GA-YSA algoritması ile geleneksel geri yayımlı YSA'ya göre daha başarılı sonuçlar elde etmişlerdir. Rangel ve diğerleri (2017) ise kısa dönemli içme suyu talep tahmininde girdi sayısı, gizli nöron sayısı, başlangıç ağırlık değerlerini optimize etmek üzere GA temelli bir YSA önermişlerdir. Son olarak Xu ve He (2018), karbondioksit miktarını tahmin etmek üzere GA'yı öğrenme oranı ve momentum faktörünü dinamik olarak optimize etmek için kullanmışlar ve deneylerle algoritmalarının etkinliğini göstermişlerdir.

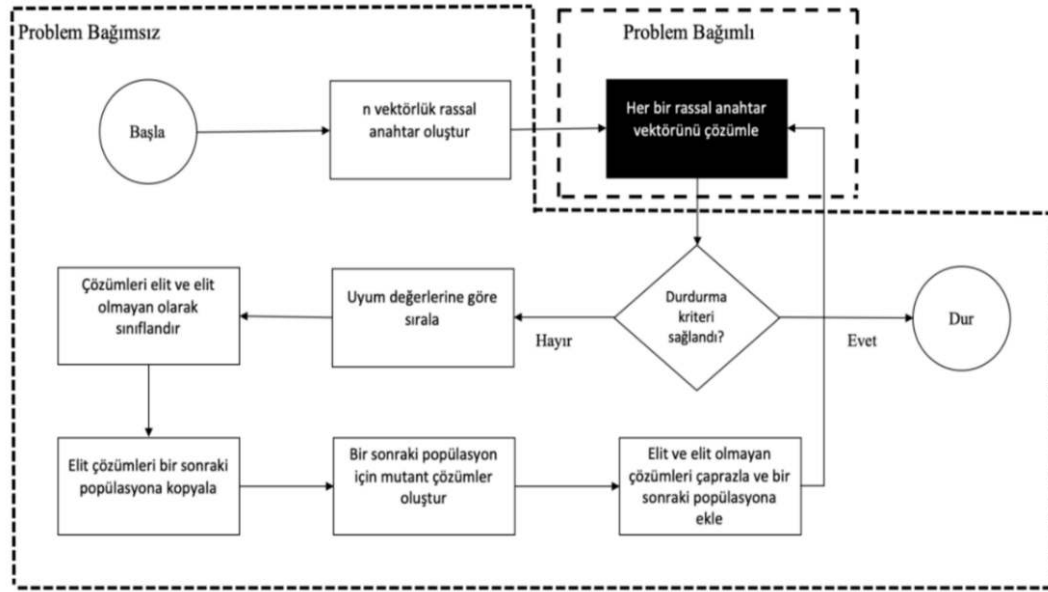
Literatürdeki çalışmalardan da görüleceği üzere, GA, YSA optimizasyonunda sıklıkla kullanılan ve başarılı sonuçlar sağlayan algoritmalarından biridir. Klasik tahmin yöntemleri ve YSA ile karşılaştırıldığında, GA çoğunlukla daha iyi sonuçlar elde edebilmektedir. Bu sebeple çalışmada trafik akışını tahmin etmek için YSA, GA'nın daha başarılı sonuçlar elde edilebilen bir çeşidi olan Yanlı Rastgele Anahtarlı Genetik Algoritma (YRAGA) seçilmiştir. YSA optimizasyonunda daha önce YRAGA kullanılan bir başka çalışmaya rastlanılmamıştır. Ayrıca, YRAGA'nın kromozom kodlama kolaylığı ve esnekliği bu metasezgiselin seçilmesinin sebeplerinden biridir.

2.4.1. Yanlı rastgele anahtarlı genetik algoritmalar

Rastgele Anahatarlı Genetik Algoritmalar (RAGA) Bean (1994) tarafından çoklu makine çizelgeleme, kaynak tahsisi ve karesel atama problemi de dahil olmak üzere çok çeşitli sıralama ve optimizasyon problemlerini ele almak için geliştirilmiştir. RAGA çözümleri kodlamak için rasgele anahtarlar kullanır. Yanlı bir rastgele anahtarlı genetik algoritma (YRAGA), ebeveynlerin çaprazlama için seçildiği şekli olarak RAGA'dan farklıdır. Bir YRAGA'da, her eleman, mevcut popülasyonda seçkin etiketli bölümden rastgele seçilen bir elemanın ve seçkin olmayan bir tanesinin çaprazlanması ile oluşturulur. Böylece her çaprazlamada belirtilen bir seviyeye göre iyi çözüm veren özelliklerin belirli bir olasılıkla bir sonraki popülasyona taşınmasına olanak verilir. Rastgele mutasyon aracılığı ile de yerel optimale takılma problemi aşılmıştır. YRAGA birçok sürekli ve kesikli optimizasyon probleminin çözümünde kullanılmıştır. Gonçalves ve Resende (2011), YRAGA'nın standart GA'dan daha iyi olduğunu göstermek amacıyla birçok kesikli ve sürekli optimizasyon problemi için örnekler vermişlerdir. Proje

çizelgeleme, makine çizelgeleme, montaj hattı dengeleme, paketleme, rotalama, ağ tasarımı, ağırlık belirleme problemleri bu problemlerden bazılarıdır.

YRAGA problem bağımlı ve problem bağımsız olmak üzere iki kısımdan oluşur. Algoritmanın problem bağımlı kısmında kromozomun ilgilenilen problemin çözümüne dönüştürüldüğü çözümleyici yer alır. Problem bağımsız kısımda ise YRAGA algoritması adımları uygulanır. Algoritmanın akış şeması Şekil 2.6’da verilmiştir.



Şekil 2.6. Yanlı rastgele anahtarlı genetik algoritmanın akış şeması

2.4.1.1. Başlangıç çözümü

N popülasyon büyüklüğünü, m ise kromozom uzunluğunu ifade etmektedir. Başlangıç popülasyonu, m uzunluğunda N adet vektör içerir. Her bir kromozom m adet $[0,1]$ arasında rastgele sayı üretilerek oluşturulur.

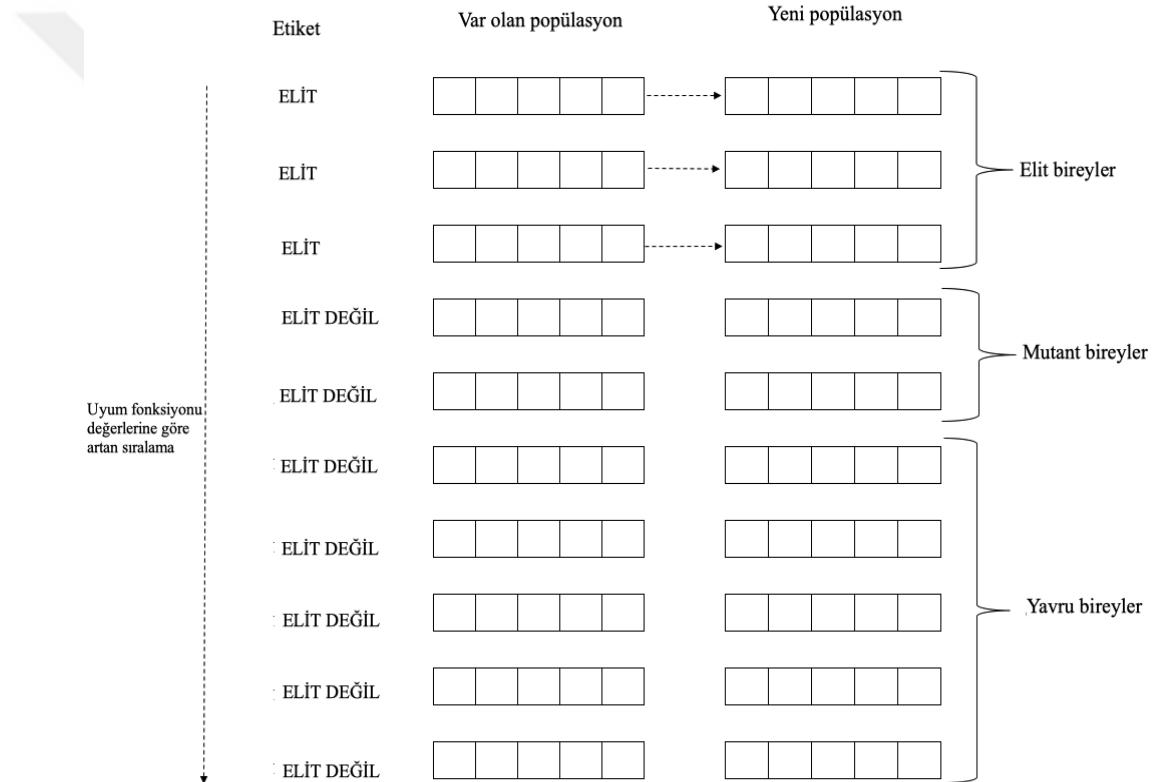
2.4.1.2. Çözücü

Kromozomda kodlanan bilginin probleme uygun hale dönüştürülmesi amacıyla çözücü (dekoder) adlı yapı kullanılır. Çözücü aracılığıyla kromozomda yer alan değerler probleme özgü olarak belirlenmiş kurallar kullanılarak problemin çözümüne uygun hale getirilir. Böylece çözümlenen kromozomlar için uyum değeri hesaplanabilir. Çözücünün probleme özgü olması ve farklı problemler için farklı çözücülerin tasarlanması YRAGA’nın uygulanmasını kolaylaştırmaktadır. YRAGA’da uygulanan çözücü harici tüm prosedürler standarttır.

2.4.1.3. Seçim ve yeni jenerasyonun oluşturulması

Probleme özgü olan uyum fonksiyonu ile uyum değeri hesaplandıktan sonra, uyum değerleri amaç fonksiyonunun yönüne göre büyükten küçüğe ya da küçükten büyüğe sıralanır. En iyi sonuç alınan belirli sayıdaki kromozom elit çözümler kümesine alınır. Popülasyonda geri kalan çözümler ise elit olmayan çözümler olarak adlandırılır.

Yeni jenerasyonda elit çözümler kümesinde yer alan kromozomlar doğrudan kopyalanır, belirli sayıda mutant çözüm oluşturulur. Jenerasyonun geri kalan kısmı ise çaprazlama ile oluşturulur. Yeni jenerasyon oluşturma süreci Şekil 2.7’de görselleştirilmiştir.



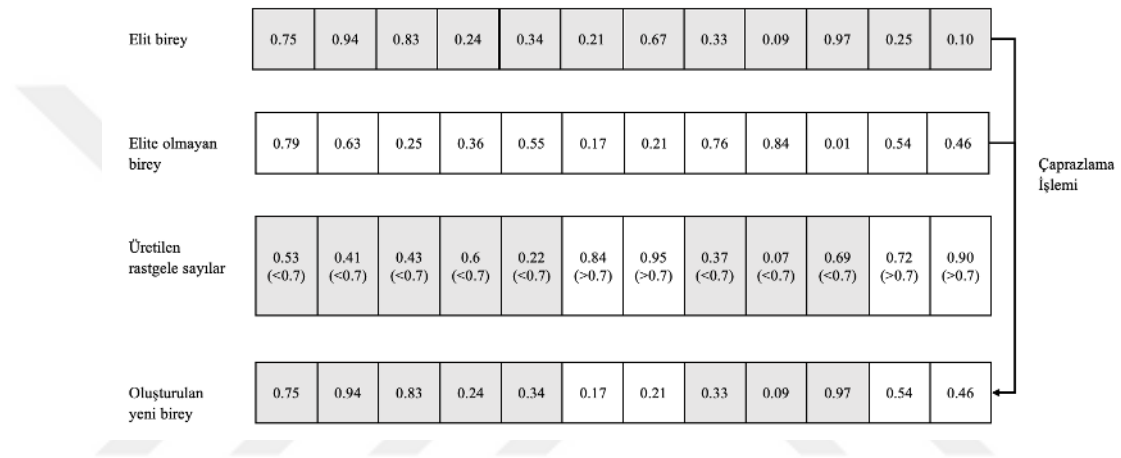
Şekil 2.7. Yanlı rastgele anahtarlı genetik algoritmada seçim ve yeni jenerasyonun oluşturulması

2.4.1.4. Mutasyon

YRAGA’da mutant çözümler, başlangıç çözümünde olduğu gibi oluşturulur. $N \times p_m$ adet m boyutlu vektör $[0,1]$ arasında üretilen rastgele sayılar ile doldurulur.

2.4.1.5. Çaprazlama

YRAGA'da çaprazlama elit kromozomlardan rastgele olarak seçilen bir ebeveyn ile elit olmayan kromozomlardan rastgele olarak seçilen bir ebeveyn aracılığıyla gerçekleşir. Belirlenen bir elit gen, seçme olasılığına göre elit çözümün özelliklerini yeni kromozoma taşır. Çaprazlama operatörü Şekil 2.8'de verilmiştir. Şekil 2.8'de görüldüğü üzere her bir gen için $[0,1]$ aralığında rastgele sayı oluşturulur. Oluşturulan rastgele sayı, önceden belirlenen çaprazlama oranı ρ_e 'den küçükse elit bireyin ilgili geni, değilse elit olmayan bireyin ilgili geni yavruya aktarılır.



Şekil 2.8. Yanlı rastgele anahtarlı genetik algorithmada çaprazlama süreci

Çaprazlama oranı ρ_e 'nin seçimi oluşturulacak yavrunun ebeveynlerinden hangisinden daha çok gen aktarılacağını etkiler. ρ_e 'nin daha büyük seçilmesi, elit ebeveynin genlerinin oluşturulan yavruya daha fazla oranda aktarılması sağlar. Böylece bahsedilen çaprazlama sonucunda elit bireyin özelliklerinin yeni jenerasyona daha fazla aktarılması sağlanır.

Popülasyonun değerlendirilmesi ve yeni jenerasyon oluşturma süreçleri belirlenen durdurma kriteri sağlanana kadar devam eder. Durdurma kriteri en fazla iterasyon sayısı, iyileşme elde edilmeyen iterasyon sayısı vb. seçilebilir.

2.4.2. Trafik akış tahminlemesi için yanlı rastgele anahtarlı genetik algoritma temelli yapay sinir ağı algoritması

Bu çalışmada ağ mimarisi ile eğitim süreci eş zamanlı olarak optimize edilmeye çalışılmıştır. Önerilen algoritma ile bir girdi katmanı, bir gizli katman ve bir çıktı katmanından oluşan bir YSA'da yer alması gereken gizli nöron sayısı belirlenmiş, gizli

nöronlara karşı gelen eşik değerlerine ait ağırlıklar ile tüm ağdaki nöronlar arasındaki bağlantılara ait ağırlık değerleri hesaplanmıştır.

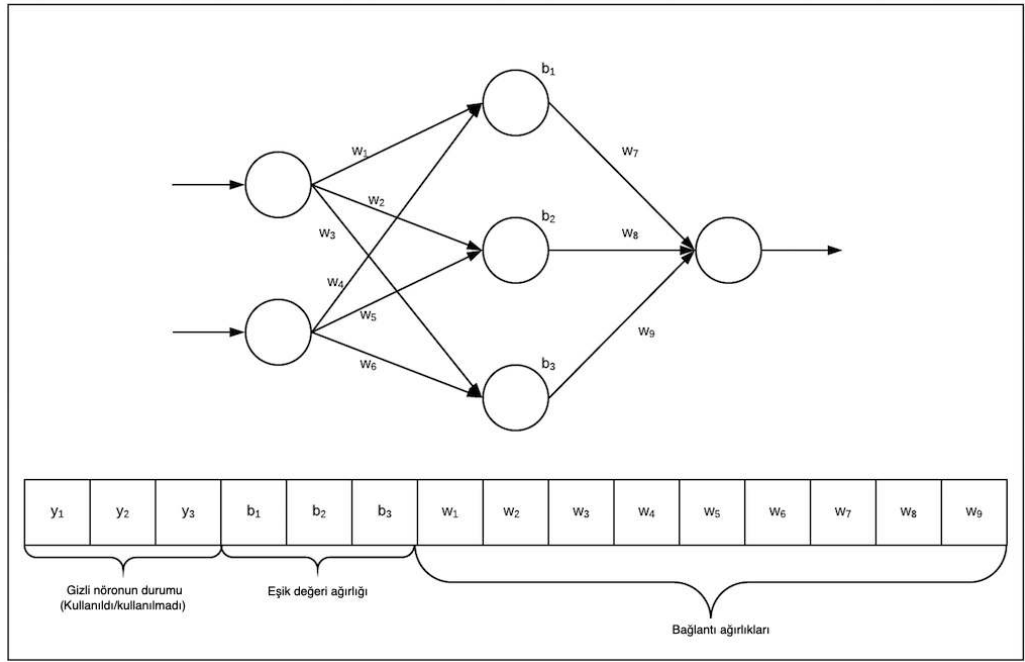
Önerilen YRAGA-YSA algoritmasında kodlanan bir kromozom 3 tür bilgiyi taşır. Bu bilgiler sırasıyla gizli nöronun kullanıp kullanılmayacağı, eşik ve ağırlık değerleridir. Başlangıçtaki ilk h gen, gizli nöronun kullanıp kullanılmayacağını gösteren 0-1 değişkendir. “ h ” en fazla kullanılması planlanan gizli nöron sayısıdır. Kromozomda yer alan ikinci h adet gen ise gizli nöronların eşik değerlerini içerir. Kromozomun kalan kısmında ise tüm ağırlıklar sırasıyla yer almaktadır.

Kromozomun boyutu kullanılacak maksimum gizli nöron sayısı ile girdi katmanından verilecek girdi sayısına bağlı olarak değişir. Kromozom uzunluğu Denklem (2.25)’te verildiği şekilde hesaplanmaktadır:

$$KromozomUzunluğu = 3 \times Gizli\ nöron\ sayısı + (Girdi\ sayısı \times Gizli\ nöron\ sayısı) \quad (2.25)$$

Gonçalves ve Resende’nin (2011) önermiş olduğu gibi popülasyon büyüklüğü kromozom boyutuna bağlıdır. $c \in \mathbb{R}^+$ olmak üzere, “popülasyon büyüklüğü $c \times$ kromozom büyüklüğü” olarak hesaplanır. Ancak, c değeri hesapsal işlem boyutunu fazlasıyla etkilediğinden deneyler esnasında 2 olarak ele alınmıştır.

Şekil 2.9’da en fazla 3 gizli nöron ve 2 girdi içeren tek çıktı bir sinir ağı örneği verilmiştir.



Şekil 2.9. Üç gizli nöron ve 2 girdi nöronuna sahip üç katmanlı bir YSA ve bu mimariye ait kromozom örneği

Bu şekilde oluşturulmuş bir sinir ağı için oluşturulan sırasıyla çözümlenmemiş ve çözümlenmiş kromozomlar Şekil 2.7’de verilmiştir.

Kromozomları değerlendirebilmek için öncelikle kromozomların probleme uygun hale getirilmesi yani çözümlenmesi gerekmektedir. Bu sebeple probleme özgü önerilen çözücü, kromozomda yer alan 3 adet bilginin probleme uygun hale getirilmesi için çalışır. Kromozomun her bir bölümünde farklı kurallara göre kromozomu çözümler. Bu amaçla uygulanan 3 kural aşağıdaki gibi açıklanabilir:

Kural 1: İlk h adet genin değerlerinin 0/1 olması gerektiğinden gendeki değer 0.5’ten büyük ise 1, değilse 0 olarak güncellenir.

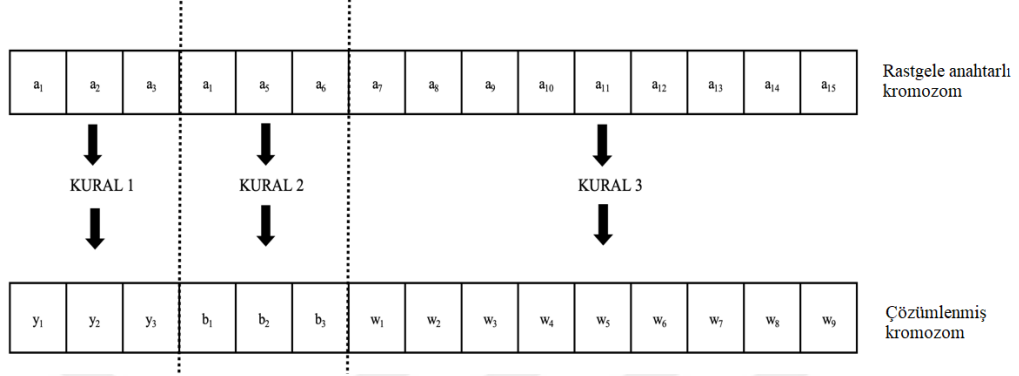
Kural 2: Eşik değerinin ağırlıklarına karşı gelen h adet genin değeri için; Denklem 2.26’da verildiği şekilde hesaplama yapılır:

$$b_i = y_i \times a_{i+h-1} \quad (2.26)$$

Kural 3: Kromozomun ağırlıkları temsil eden son kısmı için k , $\{-1,1\}$ kümesinden bir rastgele sayı olmak üzere her bir bağlantının ağırlık değeri Denklem (2.27)’deki gibi belirlenir:

$$w_j = y_j \times a_{j+2h-1} \times k \quad (2.27)$$

Böylece rastgele olarak ağırlığın $[-1,1]$ aralığında değer alması sağlanır. Şekil 2.10'de verilen YSA için çözücünün uygulanma şekli verilmiştir.



Şekil 2.10. Kromozomun çözümlenmesi

Oluşturulan kromozomların değerlendirilebilmesi için uyum fonksiyonu olarak MSE seçilmiştir. Bir j kromozomun uyum değeri, M birimlik bir veri setindeki her bir gerçek y_i için elde edilen tahmin değeri $\hat{y}_i$ olmak üzere Denklem (2.28)'deki gibi hesaplanır:

$$Uyum_j = \frac{1}{M} \sum_{i=1}^M (y_i - \hat{y}_i)^2 \quad (2.28)$$

Mutasyon ve çaprazlama işlemleri için YRAGA adımları uygulanmıştır. Bir sonraki popülasyona aktarılacak elit birey oranı p_e %20, oluşturulan her yeni popülasyon için mutant birey oranı p_m ise %20 olarak alınmıştır. Popülasyonun geri kalan %60'ı çaprazlama sonucunda oluşturulmuştur. Çaprazlama esnasında genin seçkin ebeveynden gelme olasılığı $\rho_e = 0.7$ olarak belirlenmiştir.

YSA modeli için daha önce belirtildiği gibi bir girdi, bir gizli ve bir çıktı katmanından oluşan üç katmanlı bir model ele alınmıştır. Ele alınan modelde gizli nöronlarda kullanılacak aktivasyon fonksiyonu ReLU fonksiyonu seçilmiştir. ReLU fonksiyonu formülasyonu Denklem (2.29)'te verilmiştir.

$$f(x) = \max(0, x) \quad (2.29)$$

Algoritmanın eğitim aşamasında durdurma kriteri maksimum iterasyon sayısı olarak belirlenmiştir. Maksimum iterasyon sayısı 1000 olarak verilmiştir. Ancak, overfitting problemlerinden kaçınmak için farklı durdurma kuralları da belirlenmiştir. Bu kurallardan ilki, erken durdurmadır. Uyum fonksiyonu değerinde en az %20 kötüleşme tespit edilip, uyum fonksiyonu değerinde kötüleşme devam ederse algoritma durdurulur. Ek olarak, uyum fonksiyonu değerinde arka arkaya herhangi bir iyileşme olmayan iterasyon sayısı 100 olduğunda algoritma sonlandırılır.

Geliştirilen YRAGA-YSA algoritması adımları Tablo 2.2’de verilmiştir.



Tablo 2.2. YRAGA-YSA algoritması adımları

Girdi $n, p_e, p_m, \rho_e, N_{gizli}, N_{girdi}, N_{çikti}, maksimumİterasyon$	
Çıktı Gizli nöron sayısı, gizli nöronların eşik değeri ağırlığı, bağlantı ağırlıkları	
1:	Başla
2:	$iterasyon = 0$
3:	Popülasyonu başlat: n vektörlük rastgele anahtar oluştur
4:	Eğer $iterasyon < maksimumİterasyon$ ise:
5:	Popülasyonun her vektörünü çözümle
6:	Her bir vektör için eğitim veri setini kullanarak ileri doğru hesaplama ile tahmin yap
7:	Her bir vektör için uyum fonksiyonu değerlerini hesapla
8:	Uyum değerlerini artan şekilde sırala
9:	İlk np_e vektörü “ELİT” olarak etiketle
10:	Kalanları “ELİT OLMAYAN” olarak etiketle
11:	np_m adet mutant birey oluştur
12:	ELİT ve ELİT OLMAYAN bireylerle çaprazlama ile yavruları oluştur
13:	Popülasyonu ELİT, yavru ve mutant bireylerle güncelle
14:	$iterasyon += 1$
15:	Popülasyonun her vektörünü çözümle
16:	Her bir vektör için doğrulama veri setini kullanarak ileri doğru hesaplama ile tahmin yap
17:	Her bir vektör için uyum fonksiyonu değerlerini hesapla
18:	Uyum değerlerini artan şekilde sırala
19:	İlk sıradaki vektör ile test veri setini kullanarak ileri doğru hesaplama ile tahmin yap
20:	MSE hesapla
21:	Bitiş

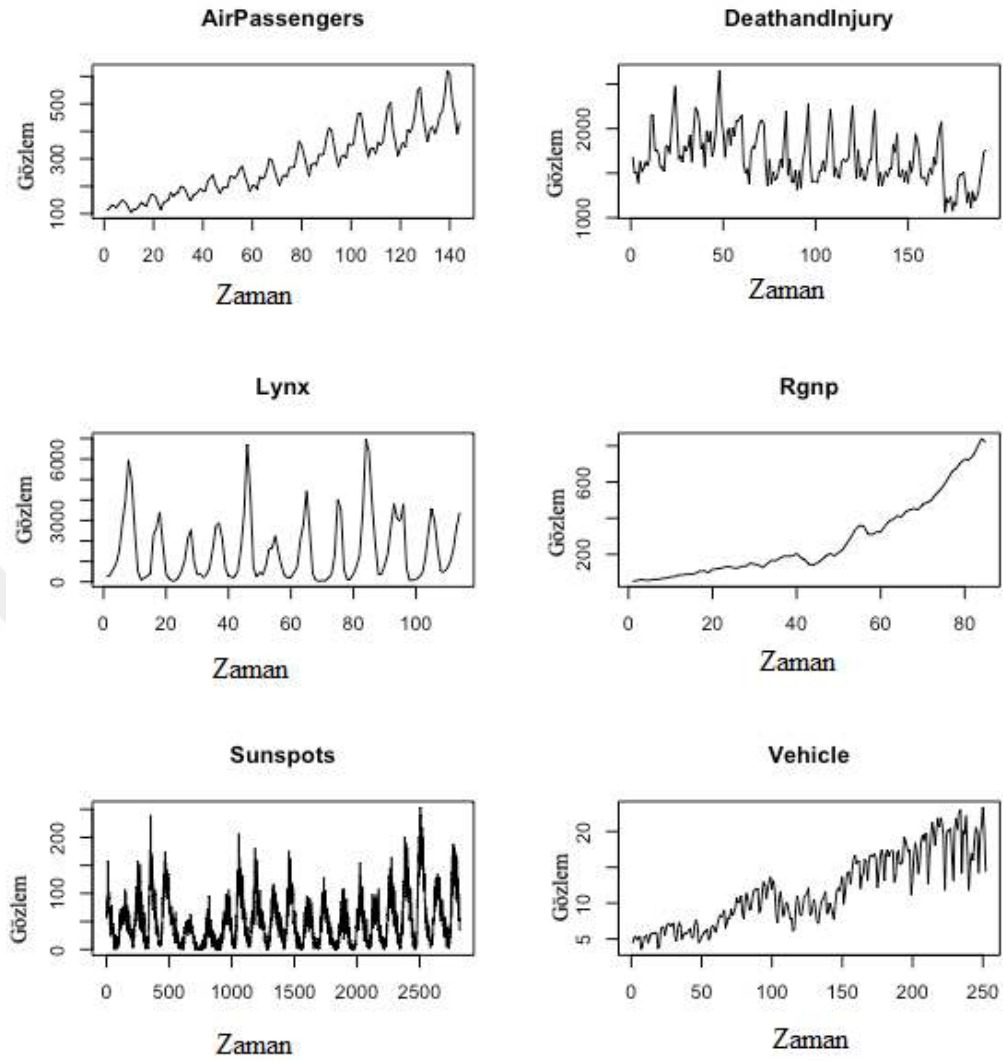
2.4.3. Uygulama ve veri setleri

Önerilen melez algoritmayı test etmek üzere Zaman Serileri Veri Kütüphanesi’nden (Hyndman, 2018) sıklıkla kullanılan bazı veri setleri seçilmiştir. Bu veri setlerine ait bilgiler Tablo 2.3’te özetlenmiştir.

Tablo 2.3. Seçilen zaman serisi veri setleri

Veri seti	Açıklama	Tür	Büyüklik
Airpassengers	Aylık uluslararası uçak yolcuları (Ocak 1949-Aralık 1960 arası)	Mevsimsel	144
Deathandinjury	İngiltere’de gerçekleşen trafik kazalarında toplam ölü ve yaralı sayısı (Ocak 1969-Aralık 1984 arası)	Mevsimsel	384
Lynx	MacKenzie nehrinde yakalanan yıllık vaşak sayısı (1821-1934 arası)	Durağan, mevsimsellik yok	114
RGNP	Amerika’da reel gayri safi mili hasıla (1800-1974 arası)	Durağan, mevsimsellik yok	85
Sunspots	Güneşteki leke sayısı (1700-1988 arası)	Durağan, mevsimsellik yok	2820
Vehicle	Amerika’da petrol ve ilgili ürünlerin (kimyasal, kömür, araç) sayısı (1971-1991 arası)	Durağan, değil, mevsimsellik yok	252

Veri setlerinin yapısını incelemek amacıyla veri setleri Şekil 2.11’de görselleştirilmiştir.



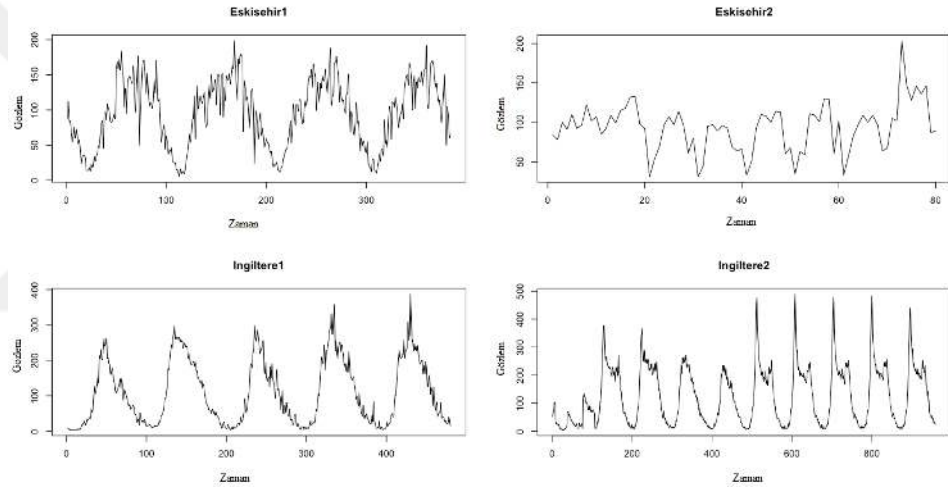
Şekil 2.11. Test edilen veri setleri

Çalışmanın ana konusu olan trafik akış tahminlemesi için elde edilen Eskişehir trafik akış verisetlerinden iki adet veri seti oluşturulmuştur. Ayrıca İngiltere trafik akış sayımları (Highways England, 2018) açık kaynak olarak paylaşılmaktadır. Elde edilen farklı trafik akış karakteristiklerine sahip ve daha büyük boyutlu olan bu verilerden de iki adet veri seti elde edilmiştir. Ele alınan tüm trafik akış verisetlerinde trafik akış sayımları 15 dakikalık periyodlar halinde toplanmıştır. Trafik akış sayımı tahmini için oluşturulan veri setleri ile ilgili bilgiler Tablo 2.4’te verilmiştir.

Tablo 2.4. *Trafik akış veri setleri*

Veri seti	Açıklama	Büyüklik
Eskisehir1	ELMS kavşağında batı-batı yönünde hareket eden araçların 4 tam günlük araç sayımı	384
Eskisehir2	Trafo kavşağında batı-güney yönünde hareket eden araçların 8 günlük 16:30-19:00 arası araç sayımları	80
İngiltere1	5 tam günlük trafik akışı	480
İngiltere2	10 tam günlük trafik akışı	960

Trafik akış veri setlerinin yapısı Şekil 2.12’de verilmiştir.



Şekil 2.12. *Trafik akış veri setleri*

Daha önce belirtildiği gibi trafik akışının mevsimsellik içerdiği ve bir günlük bir döngü içerdiği Şekil 2.12’de de açıkça görülmektedir.

Tahminler gerçekleştirilmeden önce ele alınan veri setlerine Denklem (2.30)’da verilen minmax normalizasyonu uygulanmıştır.

$$z_i = \frac{x(i) - \min(x)}{\max(x) - \min(x)} \quad (2.30)$$

Denklem (2.30)’daki z_i , x veri setindeki x_i ’nin normalleştirilmiş halidir.

Makine öğrenmesi algoritmaları ile bir zaman serisi tahmin modeli oluşturmak için öncelikle zaman serisinden bir veri seti oluşturulmalıdır. Veri seti normleştirildikten sonra, tahmin edilecek zaman için istenilen değer hedef değeri olmak üzere; bu değeri tahmin etmek için kullanılacak geçmiş veri ise hedef değere ait girdi olarak ifade edilir. Örneğin, bir t anı için yapılacak tahmin bir önceki dönem verisi ile sağlanacaksa bu durumda tahmin modelinin girdisi y_{t-1} , çıktısı ise y_t olacak şekilde tüm veri seti tahmin modeline uygun hale getirilir. Tablo 2.5'te n dönem önceki geçmiş veriden itibaren oluşturulan bir tahmin modeli için veri setinin oluşturulma şekli verilmiştir.

Tablo 2.5. Zaman serisi tahminleme veri setinin oluşturulması

Girdi	Çıktı
$y_{t-2n}, \dots, y_{t-n-2}, y_{t-n-1}$	y_{t-n}
...	...
$y_{t-n-1}, \dots, y_{t-3}, y_{t-2}$	y_{t-1}
$y_{t-n}, \dots, y_{t-2}, y_{t-1}$	y_t

Bu çalışmada da girdi sayısını belirlemek üzere YRAGA-YSA algoritmasında girdi nöronu sayısı $[1,10]$ aralığında değiştirilerek en iyi değeri veren girdi sayısı bulunmuştur. Diğer modeller için de elde edilen bu girdi sayıları esas alınmıştır.

Tahmin performansını ölçmek amacıyla literatürde en çok kullanılan performans metrikleri olarak ortalama mutlak hata (MAE), hata kareleri ortalaması (MSE) ve ortalama mutlak hata yüzdesi (MAPE) seçilmiştir. Performans metrikleri sırasıyla Denklem (2.31), (2.32) ve (2.33)'te verilmiştir.

$$MAE = \frac{1}{N} \sum_{k=1}^N |F_k - A_k| \quad (2.31)$$

$$MSE = \frac{1}{N} \sum_{k=1}^N (F_k - A_k)^2 \quad (2.32)$$

$$MAPE = \frac{1}{N} \sum_{k=1}^N \frac{|F_k - A_k|}{A_k} \quad (2.33)$$

Denklem (2.31), (2.32) ve (2.33)'teki F_k ve A_k sırasıyla N adet veri içeren bir zaman serisi veri setindeki k . örneğin tahmin ve gerçek değerlerini ifade eder.

2.4.4. Algoritma parametrelerinin seçimi

YRAGA-YSA algoritmasının parametreleri Gonçalves ve Resende (2011) tarafından önerilmiş olan parametre aralığının bir kısmı için uygulanan grid search tekniği ile belirlenmiş ve en iyi tahmin değerlerini veren parametre kombinasyonu Tablo 2.6'da verilen parametre değerlerinin arasından seçilmiştir.

Tablo 2.6. *Parametre değerleri*

Parametre	Değer
p_e	{0.1, 0.15, 0.20, 0.25}
p_m	{0.1, 0.15, 0.20, 0.25, 0.30}
ρ_e	{0.5, 0.6, 0.7}
c	{2, 3}

Seçilen parametreler Tablo 2.7'de verilmiştir.

Tablo 2.7. *YRAGA-YSA algoritması içi seçilen parametreler*

Parametre	Değer
En fazla iterasyon sayısı	1000
En fazla gizli nöron sayısı (h)	10
Popülasyon büyüklüğü (n)	$2 \times l$
p_e	$0.20 \times n$
p_m	$0.15 \times n$
ρ_e	0.7

Tüm verisetleri için YRAGA-YSA algoritması 10 kez çalıştırılmış ve elde edilen en iyi değere ait YSA modeli parametreleri kullanılarak geriye yitilmiş YSA modeli kurulmuştur.

DVR ve ARİMA/SARİMA yöntemlerinin parametreleri için grid search tekniği uygulanmıştır. Tablo 2.6'da verilen yöntemlere ait parametre değerleri arasından en iyi değeri veren modeller seçilerek ilgilenilen verisetleri için tahminler elde edilmiştir. SARİMA modelinin mevsimsellik parametresi verisetlerine uygun olarak belirlenmiştir.

Tablo 2.8. *DVR ve ARİMA/SARİMA yöntemleri için parametre değer kümeleri*

Yöntem	Parametre	Değer
DVR	Kernel	Doğrusal, RBF
	ϵ	0.01, 0.1, 0.2, 0.3
	RBF katsayısı	0.01, 0.1, 0.2, 0.3
	Ceza parametresi	1e2, 1e3
ARİMA/SARİMA	p	1, 2
	d	0, 1
	q	0, 1

2.4.5. Hesaplamalı sonuçlar

Verisetleri hazırlandıktan ve algoritmalara ait parametreler belirlendikten sonra verisetleri %70'i eğitim, %10'u doğrulama ve %20'si test olmak üzere üç parçaya bölünmüştür. YRAGA-YSA ve geriyayılımlı YSA modelleri 10'ar kez, DVR ve ARİMA/SARİMA modelleri ise grid seach için belirlenen parametrelerin denenebilmesi için her parametre kombinasyonu için birer kere çalıştırılmıştır. Kullanılan tüm tahmin modelleri Python 3.5.4 versiyonu ile kodlanmıştır. DVR modeli için Sci-kit Learn, ARİMA/SARİMA modelleri için ise Statmodels paketlerinden faydalanılmıştır. Testler, Intel(R) Core™ i7 işlemcili 2.5 GHz ve 16GB RAM'e sahip bir bilgisayarda gerçekleştirilmiştir.

Tüm testlerin sonucunda, test verisetleri için elde edilen en iyi değerler Tablo 2.8'de verilmiştir.

Tablo 2.8. Seçilen verisetleri için tahminleme sonuçları

Veri seti	Hata metriği	YRAGA-YSA	YSA	DVR	ARİMA/SARİMA
AirPassengers	MAE	23.743	39.048	23.668	46.437
	MSE	954.787	2230.295	804.304	2824.763
	MAPE	0.053	0.087	0.055	0.108
DeathandInjury	MAE	119.557	131.23	138.808	119.729
	MSE	20842.798	29419.32	32704.447	25640.153
	MAPE	0.084	0.097	0.102	0.87
Lynx	MAE	426.139	533.532	543.877	643.857
	MSE	550359.623	637203.868	570942.867	712988.96
	MAPE	0.37	0.72	0.92	1.404
RGNP	MAE	13.165	122.771	29.855	18.055
	MSE	239.278	20791.433	1236.53	487.516
	MAPE	0.021	0.179	0.045	0.028
Sunspots	MAE	13.667	18.023	17.032	13.815
	MSE	353.051	604.034	451.716	359.290
	MAPE	0.472	0.321	1.458	0.438
Vehicle	MAE	2.337	3.243	2.42	2.399
	MSE	8.913	13.228	8.757	9.262
	MAPE	0.140	0.172	0.138	0.143

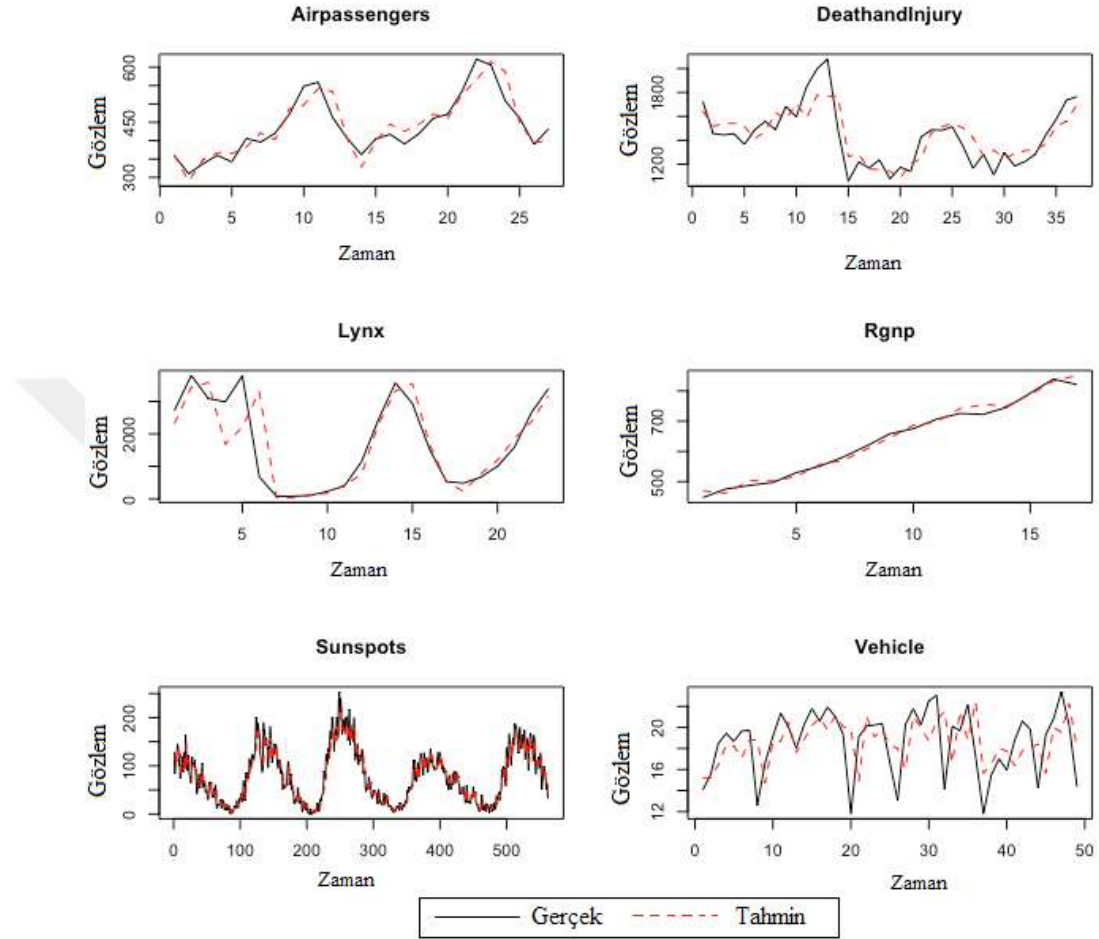
Tablo 2.8’den görüldüğü üzere, YRAGA-YSA algoritması ile diğer uygulanan algoritmalarla göre daha başarılı sonuçlar elde edilmiştir. AirPassengers veri setinde en iyi değer DVR yöntemi ile elde edilmiş olmasına rağmen, YRAGA-YSA de oldukça iyi sonuçlar vermiştir. MAPE değerlendirmesi için Lewis’in önermiş olduğu Tablo 2.9’da (Moreno vd., 2013) verilen ölçek kullanılmıştır.

Tablo 2.9. MAPE değerleri ölçeği

MAPE	Yorum
<10	Doğruluğu yüksek tahmin
10-20	İyi tahmin
20-50	Kabul edilebilir tahmin
>50	Yanlış tahmin

Bu ölçeğe göre, YRAGA-YSA tarafından AirPassengers, DeathandInjury ve RGNP veri setleri için elde edilen tahminler “doğruluğu yüksek tahmin” olarak

sınıflandırılmaktadır. Vehicle veri seti için tahminler “iyi tahmin” olarak kabul edilirken, Lynx ve Sunspots için elde edilen tahminler MAPE değerlerine “kabul edilebilir tahmin” olarak sınıflandırılmıştır. Elde edilen tahminler Şekil 2.13’te görselleştirilmiştir.



Şekil 2.13. Test verisetleri için gerçek ve tahmin değerleri

YRAGA-YSA ile elde edilen tahminlerin diğer tahminlerden istatistiksel farklı olduğunu göstermek için parametrik olmayan Wilcoxon işaret testi uygulanmıştır. Yapılan testte önerilen algoritma ile test verisetleri için elde edilen tahminlerin mutlak hata değerleri ile karşılaştırma yapılan tüm algoritmalarla test verisetleri için elde edilen tahminlerin mutlak hataları kullanılmıştır. Buna göre sırasıyla her bir karşılaştırma için aşağıdaki hipotezler kurulmuş ve uygulanan Wilcoxon işaret testi sonuçları Tablo 2.10’da verilmiştir.

H_0 : YRAGA-YSA ile ... yöntemi arasında fark yoktur.

H_a : YRAGA-YSA ile ... yöntemi birbirinden farklıdır.

Tablo 2.10. Wilcoxon işaret testi sonucu karşılaştırmalara ait elde edilen p-değerleri

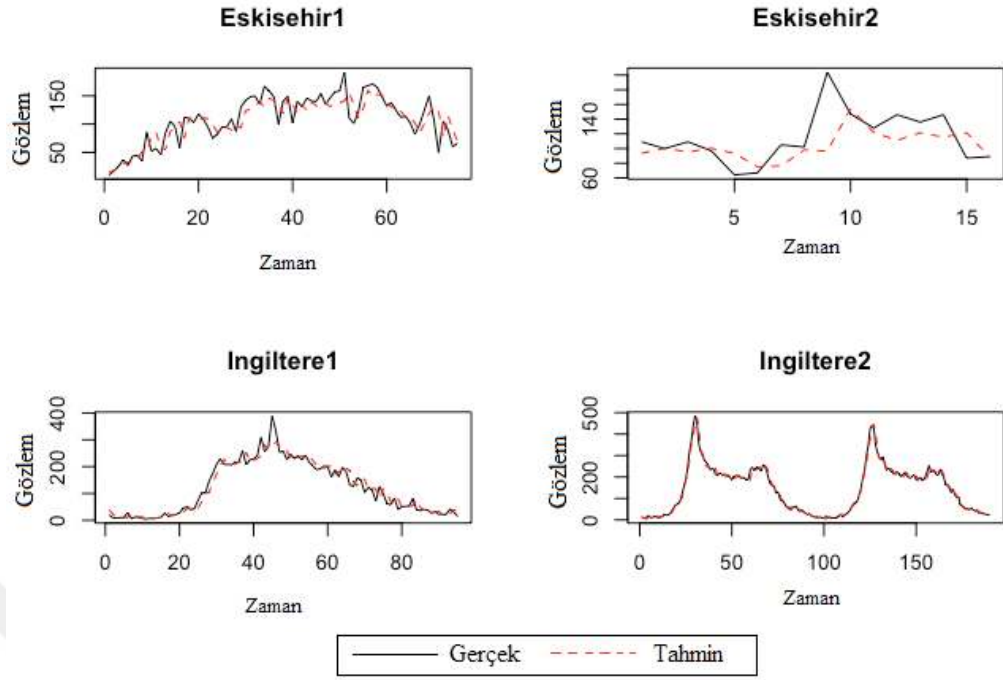
Karşılaştırılan Yöntem	p-değeri
YRAGA-YSA & YSA	0.010
YRAGA-YSA & DVR	0.000
YRAGA-YSA & ARİMA/SARİMA	0.044

Tüm testler için elde edilen p-değerleri 0.05'ten küçük olduğu için H_0 hipotezi reddedilir. Sonuç olarak, %95 güven seviyesinde YRAGA-YSA algoritması ile elde edilen tahminler diğer tüm yöntemlerle elde edilen tahminlerden farklıdır.

Trafik akış verisetleri için yapılan tahminlerin test verisetlerine ait performans değerleri ise Tablo 2.11'de verilmiştir. Ayrıca elde edilen tahmin değerleri ile gerçek değerler Şekil 2.4'te görselleştirilmiştir.

Tablo 2.11. Trafik akış verisetleri için tahminleme sonuçları

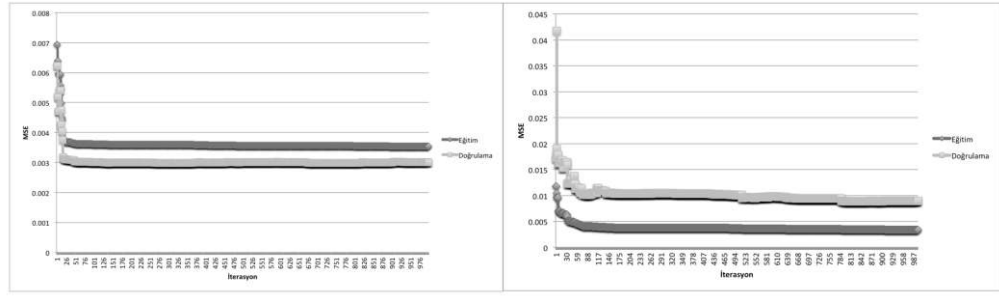
Veri seti	Hata Metriği	YRAGA-YSA	YSA	DVR	ARİMA/SARİMA
Eskisehir1	MAE	17.407	18.616	17.261	18.139
	MSE	484.990	578.232	510.266	546.235
	MAPE	0.204	0.239	0.202	0.212
Eskisehir2	MAE	21.183	22.251	22.374	26.173
	MSE	1087.492	1159.884	1179.210	1363.259
	MAPE	0.174	0.180	0.185	0.229
İngiltere1	MAE	17.195	21.539	17.448	18.023
	MSE	629.718	827.256	669.701	707.237
	MAPE	0.272	0.514	0.269	0.267
İngiltere2	MAE	10.548	11.961	8.860	11.970
	MSE	248.100	242.814	152.575	335.285
	MAPE	0.119	0.135	0.109	0.132



Şekil 2.14. Trafik akış verisetleri için gerçek ve tahmin değerleri

Trafik akış verisetleri için elde edilen tahminlerin performans değerleri incelendiğinde YRAGA-YSA ve DVR algoritmaları ile başarılı tahminler elde edildiği görülmüştür. DVR'nin daha başarılı olduğu tahminlerde de YRAGA-YSA'nın performansının DVR'ye oldukça yakın olduğu görülmüştür. Lewis'in ölçeğine göre trafik akış verisetleri için iyi ve makul tahminler elde edilebilmiştir.

Önerilen melez algoritmanın herhangi bir aşırı uyum sorunu olup olmadığını gözlemlemek için eğitim ve doğrulama verisetlerinin uygulanan iterasyonlar boyunca değişimi gözlenmiştir. Şekil 2.15'te test edilen iki veri seti için eğitim aşamasında eğitim ve doğrulama verisetlerinin uyum fonksiyonu değerleri verilmiştir. Görüldüğü üzere herhangi bir aşırı öğrenme sorunu ile karşılaşılmamıştır.



Şekil 2.15. Eğitim ve doğrulama verisetlerine ait iterasyon bazında uyum fonksiyonu değerleri

2.5. Sonuç

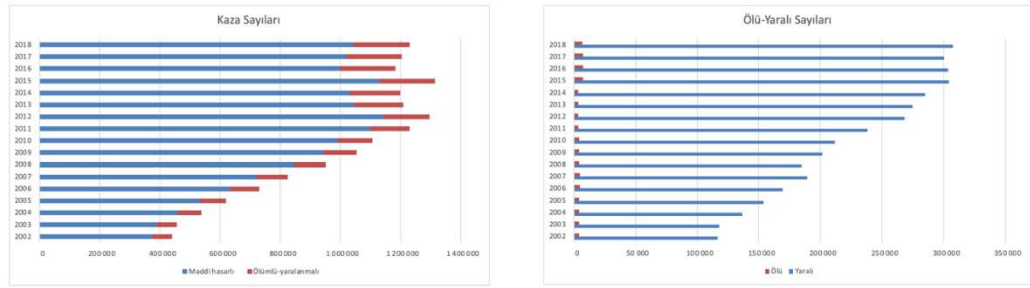
Bu bölümde bir zaman serisi olarak düşünülen trafik akışının tahminlemesi için Yapay Sinir Ağı temelli melez bir algoritma önerilmiştir. Önerilen melez algoritma geri yayımlı YSA, DVR ve ARİMA/SARİMA yöntemleri ile karşılaştırılmış ve çoğunlukla başarılı sonuçlar elde edilmiştir. Algoritmanın çalışma sürelerinin YSA mimarisini de eş zamanlı optimize etmeye çalıştığından eğitim seti büyüklüğü, girdi katmanında yer alan nöron sayısı ve en fazla gizli nöron sayısı arttıkça uzadığı görülmüştür. Bu problem melezlenen Yanlı Rastgele Anahtarlı Genetik Algoritma'nın paralel programlanabilme kolaylığı ile ortadan kolayca kaldırılabilir. Bunun yanına YRAGA-YSA algoritmasının kullanılmasının avantajları da ortaya çıkmıştır. Algoritmayı kullanmanın avantajları aşağıdaki gibi sıralanabilir.

1. Algoritmanın elde edilen eniyi sonuçtaki gizli nöron sayısını kendisinin belirlemesi,
2. Aktivasyon fonksiyonu olarak türevlenebilir bir fonksiyon kullanma zorunluluğunun ortadan kalması,
3. Algoritmanın her iterasyonda rastgele olarak oluşturulan yeni çözümleri de popülasyona dahil ettiğinden aşırı öğrenme probleminin ortadan kalması,
4. Yerel minimuma takılma tehlikesinin önüne geçilmesidir.

3. TRAFİK KAZALARININ ANALİZİ VE TAHMİNİ

Trafik kazaları trafik güvenlik seviyesini düşürürken; birçok sayıda can kaybı, yaralanma ve maddi hasarla sonuçlanmaktadır. Trafik kazaları, sürücü, yolcu, yaya, yol ve araç kusurlarından kaynaklanır. Türkiye’de gerçekleşen kazaların ortalama %92.3’ü sürücü; %6.51’i ise yaya kusurlarından kaynaklanmakta; yolcu, yol ve araç sebepli kaza oranı %1.25’lerde kalmaktadır.

TÜİK trafik kaza istatistiklerinde (Türkiye İstatistik Kurumu, 2020) belirtildiği üzere 2002- 2018 arası Türkiye’de gerçekleşen toplam 16,577,394 kazada 82,942 ölü 3,762,358 yaralı meydana gelmiştir. Gerçekleşen kazaların, ölü ve yaralı sayılarının yıllara göre dağılımları Şekil 3.1’de verilmiştir.



Şekil 3.1. Türkiye’de gerçekleşen kazalar ve ölü-yaralı sayıları

Bu ağır bilanço, trafik güvenliği seviyesinin arttırılmasına yönelik çalışmalar yapılması gerekliliğini göstermektedir. Kazaların sebeplerini anlamak gerçekleşebilecek kazaları önleyebilmek için oldukça önemlidir.

Bu amaçla, dünyanın farklı yerlerindeki birçok şehirde trafik kazaları ile ilgili çalışmalar yapılmaktadır. Kaza faktörlerinin, sürücülerin, yol parametrelerinin kazaya etkisi araştırılmış; kazalar gerçekleşmeden tahmin edebilmek için çalışmalar sunulmuştur. Bu çalışmada da Eskişehir’de gerçekleşen kazalar üzerinde çalışılmış; kazalarla ilgili bazı tanımlayıcı analitikler sunulmuştur. Bunun yanı sıra, kaza tahmini için bir kestirimsel analitik yaklaşımı uygulanmıştır.

3.1. Literatür Taraması

Birçok araştırmacı, uzun yıllardır trafik kazaları konusunda çalışmaktadır. Literatürdeki çalışmaların çoğu gerçek hayatta gerçekleşen kaza verilerini işlemişlerdir. Araştırmacılar zamansal, hava durumu, yol ve nadiren insan faktörleri gibi değişkenleri

kullanarak yaralanma şiddeti, kaza sayısı, kaza riski, ölüm / yaralanma sayısı gibi farklı kaza değişkenlerini tahmin etmeye; kazaları analiz ederek kaza karakteristiklerini ve sebeplerini ortaya koymaya çalışmışlardır.

Trafik kazalarının tahmini için ele alınan çalışmalar, kazalarla ilgili en çok tahmin edilen değişkenler olan kaza gerçekleşme riski, kaza süresi ve kaza şiddeti üzerinden incelenmiştir. Kaza riski tahmini için Lin, Wang ve Sadek (2015) K-en yakın komşu ve Bayes ağları modellerini sunmuş ve ayrıca tüm değişkenlere ve başarılı bir öznelik seçim yöntemi önermişlerdir. Son yıllarda, makine öğrenmesi yöntemleri kaza tahmininde yaygın olarak kullanılmaktadır. Sharma, Katiyar ve Kumar (2015) ile Lv ve diğerleri (2009) kaza riski tahmin problemi için Destek Vektör Makinesi (DVM) kullanmıştır. Sharma, Katiyar ve Kumar (2015) DVM'yi çok katmanlı algılayıcı (MLP) ile karşılaştırmış ve Gaussian_kernelli DVM'nin, MLP'den ve polinom kernelli DVM'den daha iyi performans gösterdiğini raporlamışlardır. Lv ve diğerleri (2009) ise, gerçek zamanlı karayolu trafik kazası tahmininde ilk kez K-en yakın komşu yöntemini uygulamıştır. Ayrıca Yuan ve diğerleri (2017), Iowa eyaletinde 2006 – 2013 arasında gerçekleşen motorlu taşıt kazası verilerini kullanarak DVM, Karar Ağacı, Rastgele Orman ve Derin Sinir Ağı (DNN) olmak üzere dört sınıflandırma değerlendirmiş ve DNN modeliyle yüksek doğruluk değerini elde etmişlerdir. Sözü edilen çalışmalardan farklı olarak, sosyal medya verileri trafik kazalarını tahmin etmek için de kullanılabilir. Örneğin, Zhang ve diğerleri (2018), Twitter'dan toplanan sosyal medya verileriyle trafik kazalarını tespit etmek için derin öğrenme yöntemini kullanmışlardır.

Kaza oluşumunun yanı sıra, tahmin edilen diğer kaza değişkenlerinin literatürüne de kısaca değinilmiştir. Trafik kazalarının tahmin edilen değişkenlerinden biri olarak, trafik kazası süresi, kazanın meydana gelmesi ile temizlenmesi veya yoldan kaldırılması arasındaki süre olarak tanımlanmaktadır. Chung (2010) Kore Otoyol Sistemlerinde kaza süresi modellemesi için bir lojistik-lojistik hızlandırılmış arıza süresi (AFT) metrik modeli önermiştir. Ayrıca, Yu ve diğerleri (2016), YSA ve DVM'yi kaza süresinin tahmini için karşılaştırmış ve DVM modelinin YSA modelinden daha iyi sonuç verdiğini belirtmişlerdir. Alkaabi, Disssanayake ve Bird (2011) ise Abu Dabi, Birleşik Arap Emirlikleri'nde kaza süresi tahmininde gama heterojenliği olmayan bir Weibull modeli kullanmıştır.

Tahmin edilen diğer bir kaza değişkeni, kaza şiddetidir. Abdelwahab ve Abdel-Aty (2001), Chang ve Wang (2006), Alkheder, Taamneh ve Taamneh (2016) kaza şiddeti üzerine çalışmışlardır. Tüm bu çalışmalarda, bu tahmin probleminin çözümü için makine öğrenmesi modelleri önerilmiştir.

Trafik kazaları incelenmesi ve sebep olan faktörlerin belirlenmesi üzerine yapılan çalışmalar genellikle gerçek verisetleri kullanılarak uygulanmıştır. Çalışmalarda Birleşik Arap Emirlikleri, Japonya, İspanya, Etiyopya gibi ülkelere ait trafik kazası verileri kullanılmıştır. Taamneh, Alkheder ve Taamneh (2017), Abu Dhabi'de 2008 – 2013 yılları arasında gerçekleşen kaza verilerini kullanarak kazalarla ilgili tanımlayıcı istatistikler sunmuşlar, inceledikleri veri setindeki kazalar ile ilgili çeşitli kurallar ortaya koymaya çalışmışlardır. Watanabe ve Nakamura (2016), Japonya'daki anayollardaki kazaları etkileyen faktörleri belirlemek için faktör analizi ve regresyon analizi modelleri oluşturmuşlardır. Chen, Zhang ve Xu (2016), Çin'de yaşanan büyük kayıplara yol açan ölümlü yol kazalarının etkileyen faktörlerin önem sıralamasını belirlemek için faktör analizi uygulamışlardır. Molla (2016), Amerika'da Kolombiya bölgesi ve Alaska harici 48 eyalete ait kazalardaki sürücü özellikleri için Temel Bileşen Analizi ve eyaletler için hiyerarşik kümeleme uygulamıştır. Beshah ve Hill (2010) ise, Etiyopya'daki kazaların 40 özelliğini ele alarak kaza şiddetini sınıflandırmaya çalışmışlardır.

Benzer şekilde Türkiye'de de trafik kazaları konusunda çalışmalar yapılmıştır. Murat ve Şekerler (2009), Denizli ili 2004, 2005 ve 2006 yıllarına ait trafik kaza verilerini klasik ve bulanık kümeleme yöntemleriyle analiz etmiş, elde edilen küme merkezlerine yakın bölgelerdeki trafik kazalarının daha yoğun olduğu noktalar kara nokta olarak belirlenmiştir. Akgüngör ve Doğan (2010), İzmir'de yaşanan kazalar için geleceğe yönelik tahminler oluşturacak regresyon, genetik algoritma ve yapay sinir ağı modelleri oluşturmuşlardır. Yıldırım ve Özcan (2015) ise Ankara ili örneğinde trafik kazalarını azaltacak ileri beslemeli bir yapay sinir ağı modeli sunmuşlardır.

İncelenen çalışmalar doğrultusunda, literatürde çoğunlukla trafik kaza değişkenlerini tahmin etmeye ve kazaları anlamaya yönelik çalışmalar yapılmıştır. Benzer şekilde bu çalışmada da Eskişehir'de yaşanan trafik kazalarını anlamaya çalışmak için bazı istatistiksel ve veri madenciliği yöntemleri kullanılarak tanımlayıcı analitikler oluşturulmuştur. Bunun yanı sıra, kaza gerçekleşme tahmini için literatürde bir örneğine

rastlanılmayan tek sınıf sınıflandırma modeli kurulmuş ve farklı yöntemler ile karşılaştırılması üzerine çalışılmıştır.

3.2. Tahminleme Yöntemleri

İki sınıf sınıflandırma iki grup olarak verilen kümenin elemanlarını sınıflandırma işlemidir. En bilinen iki sınıf sınıflandırma yöntemlerinden bazıları Destek Vektör Makineleri, K-en yakın komşuluk yöntemi, Lojistik Regresyon, Rastgele Orman Algoritmasıdır.

Tek sınıf sınıflandırma algoritmaları ise negatif sınıfın olmadığı, yetersiz örneklediği ya da iyi tanımlanmadığında sınıflandırma modelleri kurmayı amaçlar. Tek sınıf sınıflandırma algoritmaları, tek sınıf DVM'ler ve YSA, karar ağaçları ve en yakın komşuluk gibi diğer algoritmalar. Bu çalışmada tek sınıf DVM kullanıldığı için detaylı olarak açıklanacaktır.

3.2.1. Rastgele orman algoritması

Rastgele Ormanlar (RO), ilk kez Breiman tarafından sınıflandırma ve regresyon problemleri için sunulmuştur. Genuer ve diğerleri (2010), RO'nun çalışma prensibini bir L öğrenme örneğinden önyüklemeli örnekler ve X açıklayıcı değişkenlerinin rastgele seçilen bir alt kümesini kullanarak birçok karar ağacı oluşturmak şeklinde açıklamaktadır.

RO, “değişken önemi” olarak adlandırılan bir önem puanı hesaplayarak öznitelik seçimi için de kullanılabilir. RF çerçevesinde, belirli bir değişkenin önemi için en yaygın olarak kullanılan puan, bu değişkenin gözlenen değerlerine torba dışı örneklerde (OOB_t) rastgele izin verildiğinde ormandaki bir ağacın hatasının ortalamasındaki artıştır (Genuer, Poggi ve Tuleau-Malot, 2010).

Bir X^j özneliğinin değişken önemi (VI) izleyen şekilde tanımlanır. Ormanın her ağacı t için, ilişkili OOB_t örneği düşünülün. $err\widetilde{OOB}_t$, dikkate alınan OOB_t örneği üzerindeki tek bir ağacın t hatasını belirtir. $err\widetilde{OOB}_t$ alternatif olarak regresyon için MSE değeri ve sınıflandırma için yanlış sınıflandırma oranı olarak açıklanabilir. Böylece bir X^j özneliği için VI Denklem (3.1)'de verildiği şekilde hesaplanır:

$$VI(X^j) = \frac{1}{ntree} \sum_t (err\widetilde{OOB}_t^j - err\widetilde{OOB}_t) \quad (3.1)$$

Denklem (3.1)'de belirtilen toplam, Rastgele Ormanın tüm ağaçları üzerindedir ve *n*tree ise Rastgele Ormandaki ağaçlarının sayısını belirtir (Genuer, Poggi ve Teleau-Malot, 2010).

3.2.2. K-en yakın komşuluk yöntemi

K-En Yakın Komşu (kNN), bir veri noktasının belirli sayıda k komşu veri noktasına minimum mesafesini bulmaya dayanan basit ve en yaygın sınıflandırma algoritmalarından biridir. En yakın komşuların sayısı olarak k parametresini belirlemek çoğunlukla zordur, ancak sınıflandırıcı performansı için hayati önem taşır. kNN algoritmasının kendisi oldukça basittir ve aşağıdaki adımlarla özetlenebilir (Raschka, 2015):

1. k sayısını ve mesafe metriğini seçilir.
2. Sınıflandırılmak istenen örneğin en yakın k adet komşusu bulunur.
3. Sınıf etiketi oy çokluğu ile atanır.

3.2.3. Lojistik regresyon

Lojistik Regresyon (LR), bağımlı değişkenin ikili olduğu kategorik bir değişkenin tahmini için istatistiksel bir sınıflandırma modelidir. LR'nin ana konsepti olasılığa dayanmaktadır. LR, çıkış değerini 0 ile 1 arasında sınırlayan bir maliyet fonksiyonu olarak sigmoid fonksiyonunu kullanır. Denklem (3.2), dikkate alınan Y değişkeninin p bağımsız değişkenle gerçekleşme olasılığını gösterir:

$$P(Y = 1) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_p x_p)}} \quad (3.2)$$

Denklem (3.2)'deki $\beta_0, \beta_1, \beta_2, \dots, \beta_p$ regresyon katsayılarıdır.

Denklem (3.3), Y 'nin gerçekleşme olasılığının ($P(Y = 1)$), gerçekleşmeme olasılığına ($1 - P(Y = 1)$) oranının doğal logaritması X_i 'ye bağlı doğrusal bir model verir.

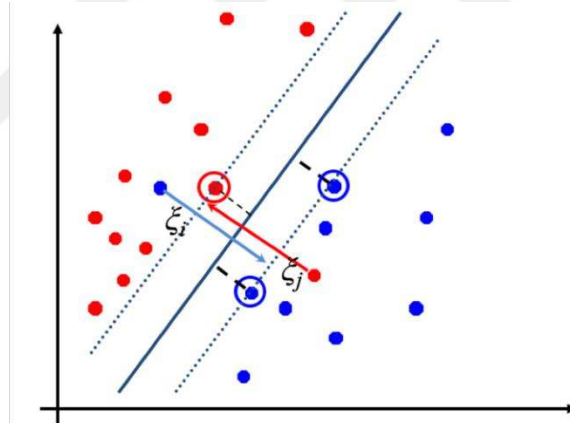
$$g(x) = \ln \left(\frac{P(Y = 1)}{1 - P(Y = 1)} \right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_p x_p \quad (3.3)$$

$g(x)$, bir doğrusal regresyon modelinin istenilen birçok özelliğine sahiptir. Bağımsız değişkenler, sürekli ve kategorik değişkenlerin bir kombinasyonu olabilir (Kurt, Türe ve Kurum, 2008).

3.2.4. Destek vektör makineleri

Sınıflandırma problemleri için en popüler makine öğrenimi yöntemlerinden biri Vapnik tarafından geliştirilen DVM'lerdir. DVM istatistiksel öğrenme teorisinden türetilen ve denetimli öğrenme yapan bir yöntemdir.

DVM'lerin amacı N boyutlu bir öznitelik uzayında veri noktalarını ayrı ayrı sınıflandıran bir hiperdüzlem bulmaktır. İki sınıfa ait verileri sınıflandırmak için birçok hiperdüzlem seçilebilir. Ancak DVM bu hiperdüzlemler arasından maksimum marja sahip olan hiperdüzlemi bulur. Bu optimizasyon probleminde marj karar sınırı olan hiperdüzlem ile veri setindeki herhangi bir örnek arasındaki en kısa uzaklık olarak tanımlanır. DVM'lerin temel fikri Şekil 3.2'de görselleştirilmiştir.



Şekil 3.2. DVM'lerin çalışma prensibi

Algoritmik olarak, DVM'ler kısıtlı kuadratik optimizasyon problemini çözerek veri kümeleri arasında optimal ayırma sınırlarını oluşturur (Dreiseitl ve Ohno-Machado, 2002). Doğrusal, polinom ve RBF gibi kernel fonksiyonlar kullanılarak, doğrusal olarak ayrılamayan veriler DVM ile başarıyla sınıflandırılabilir.

3.2.5. Tek sınıf destek vektör makineleri

Schölkopf ve diğerleri, tek sınıf sınıflandırma problemi için tek sınıf DVM'yi, DVM'nin bir uzantısı olarak sunmuşlardır. $\{x_i, i = 1, 2, \dots, n\}$ 'nin eğitim örnekleri olduğu varsayılarak bir $\varphi(\cdot)$ fonksiyonu bir x_i örneğini X girdi uzayından F öznitelik uzayına

eşleştirir. Tek sınıf DVM'nin temel fikri, tüm veri noktalarının başlangıç noktasının karşı tarafında yer aldığı ve marjın pozitif olacağı şekilde, başlangıç noktasından maksimum uzaklıkta olan bir hiper-düzlemin oluşturulmasıdır (Xiao, Wang ve Xu, 2015). Tek sınıf DVM'nin matematiksel modeli Denklem (3.4) ve (3.5)'te verilmiştir.

$$\min_{w \in F, \xi \in \mathbb{R}^n, \rho \in \mathbb{R}} \frac{1}{2} \|w\|^2 + \frac{1}{vn} \sum_i \xi_i - \rho \quad (3.4)$$

$$\text{s.t. } \langle w, \varphi(x_i) \rangle \geq \rho - \xi_i, \quad \xi_i \geq 0 \quad (3.5)$$

Denklemlerde w hiperdüzlemin normal vektörü, ρ eşik değeri ve ξ_i ise aylak değişkendir. Düzenleme terimi v kullanıcı tarafından verilen, ödünleşimi kontrol eden ve tanım dışındaki kabul edilebilir örnek oranını gösterir. (Zhou vd., 2005).

Tek sınıf DVM probleminin Lagrange ikili, Denklem (3.6), (3.7) ve (3.8)'de verilmiştir.

$$\max_{\alpha} -\frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j k(x_i, x_j) \quad (3.6)$$

$$\text{s.t. } 0 \leq \alpha_i \leq \frac{1}{vn} \quad (3.7)$$

$$\sum_{i=1}^n \alpha_i = 1 \quad (3.8)$$

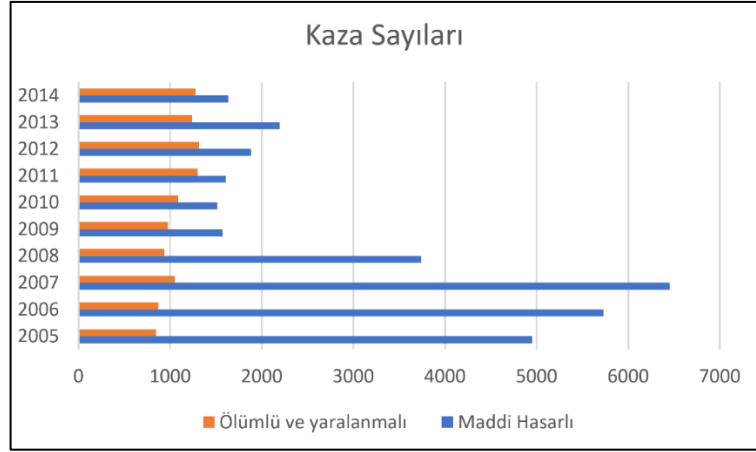
Burada $k(x_i, x_j)$ kernel fonksiyonudur. Kernel fonksiyonu ile amaç fonksiyonu tek sınıf DVM'nin Denklem (3.9)'da verilen karar fonksiyonu olan $f(x)$ 'e dönüşür.

$$f(x) = \langle w, \varphi(x) \rangle - \rho = \sum_{i=1}^n \alpha_i k(x_i, x) - \rho \quad (3.9)$$

Eğer $f(x) \geq 0$ olursa x_i örneği hedef olarak etiketlenirken; diğer durumlarda hedef olmayan şeklinde etiketlenir (Xiao, Wang ve Xu, 2015).

3.3. Kaza Verileri

Eskişehir Akıllı Trafik Kontrol Merkezi'nden alınan veriler doğrultusunda Eskişehir'de 2005 ile 2014 yılları arasında gerçekleşen ölümlü-yaralanmalı ve maddi hasarlı kazalar bilgileri incelenmiştir. Şekil 3.3'te gerçekleşen kaza sayıları verilmiştir.



Şekil 3.3. Eskişehir’de gerçekleşen kaza sayıları

Kaza verileri kaza raporlarından elde edilen kaza bilgileri, kaza zamanı ile ilgili bilgiler, kazanın yaşandığı yerle ilgili bilgiler ve kazaya karışan araç ile araç sürücüsü hakkında bilgiler içerir. Ölümlü-yaralanmalı kazalar ek olarak kazaya karışan yaya veya yolcu bilgilerini de içermektedir. Tablo 3.1’de ölümlü-yaralanmalı ve maddi hasarlı kazalara ait öznitelikler özetlenmiştir.

Tablo 3.1. Kaza öznitelikleri

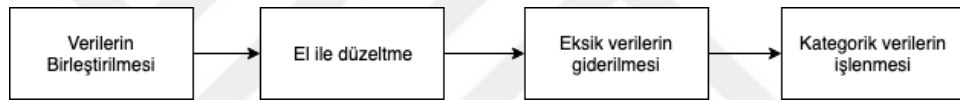
Kategori	Öznitelik	Değer
Kaza bilgileri	Kaza türü	Tek araçlı, zıt yönlü iki araçlı, aynı yönlü iki araçlı, çok araçlı
	Çarpışma türü	Park halindeki araca çarpma, yayaya çarpma, yandan çarpışma, arkadan çarpma, ...
Zamansal bilgiler	Kazaya karışan araç sayısı	≤4
	Ölü sayısı	>0
	Yaralı sayısı	>0
	Maddi hasar miktarı	TL
	Saat	Saat: Dakika
	Tarih	Gün/Ay/Yıl
	Haftanın günü	Pazartesi, Salı, Çarşamba, Perşembe, Cuma, Cumartesi, Pazar
	Mevsim	İlkbahar, Yaz, Sonbahar, Kış
	Gün durumu	Gündüz, akşam, gece
	Hava durumu	Açık, bulutlu, sisli, yağmurlu, karlı, fırtınalı
Kaza yeri bilgileri	Yol yüzeyi	Kuru, ıslak, buzlu, tuzlu, çamurlu, yağ veya akaryakıt dökülmüş
	Yol türü	Ana yol, il yolu, devlet karayolu, sokak, kavşak, diğer
	Yol bölümü	Bölünmüş/bölünmemiş
	Şerit sayısı	≥ 1
	Şerit çizgileri	Var:1, Yok:0
	Yolun kaplama cinsi	Asfalt, beton, parke, stabilize, ham
	Kaldırım	Var:1, Yok:0
	Banket	Var:1, Yok:0
	Viraj türü	Düz yol, hafif viraj, keskin viraj
	Geçit	Kontrollü demiryolu geçidi, kontrolsüz demiryolu geçidi, yaya geçidi, okul geçidi, ...
	Trafik ışığı	Var:1, Yok:0
	Trafik işareti	Var:1, Yok:0
	Tümsek	Var:1, Yok:0
	Yol çalışması	Var:1, Yok:0
	Kavşak türü	Döner kavşak, 3 yönlü (T-şekli), 3 yönlü (Y-şekli), ..., kavşak yok
Kişisel bilgiler	Diğer	Dar yol, kasis, menfez üstü, ...
	Sürücünün yaşı	18- 80 arası
	Sürücünün cinsiyeti	Kadın, erkek
	Sürücü belgesi türü	A, B, C, D, F, G
	Sürücünün eğitim durumu	İlkokul, ortaokul, lise, üniversite, yüksek lisans
	Araç türü	Otomobil, bisiklet, motosiklet, motorbisiklet, minibüs, otobüs, kamyon, kamyonet, traktör, tramvay, tren, iş makinesi, çekici, ...
	Kandaki alkol seviyesi	mg alkol / 100 ml kan
	Sürücünün asli kusuru	2918 nolu Karayolları trafik kanununda belirtilen kusur
	Kazaya karışan yaya/yolcu	Var:1, Yok:0

Elde edilen veri kayıtlarındaki eksiklerden dolayı 2005-2012 yılları arasındaki kazalar kullanılarak kaza verisetleri oluşturulmuştur.

3.3.1. Veri önışleme

Trafik kaza verileri kaza sonrası tutulan raporlardan elde edilmektedir. Her gerek hayat verisinde olduđu gibi veri toplama veya giriř esnasında yanlışlıklar yapılabilir. Elde edilen verilerde kaza özniteliklerinin olması gereken deđer aralıklarında ya da formatta girilmemesi, bir öznitelik bilgisinin bir başka öznitelik alanına girilmesi ya da bir alanın tamamen boş bırakılması gibi sistematik olmayan insan hataları ile karşılaşılmasıdır. Bu sebeple eksik veriler dışındaki tüm hatalar ilk aşamada el ile düzeltilmiştir.

Elle düzeltme işleminden sonra eksik veriler özniteliğın türüne göre doldurulmuştur. Eksik verilerin içinde yer aldığı kategorik ve sayısal değere sahip öznitelikler sırasıyla ilgili sütundaki en çok tekrarlayan değere ve ilgili sütunun ortalama değeri ile doldurulmuştur. Şekil 3.4'te kaza verisetleri için uygulanan önışleme adımları verilmiştir.



Şekil 3.4. Önışleme adımları

Verilerin üzerinde herhangi bir işlem yapılabilmesi için kategorik değere sahip özniteliklerin sayısallaştırılması gerekmektedir. Kategorik verilerin sayısallaştırılmasında kategoriye ait değerler doğrudan sayısal değerlere dönüştürülemez. Örneğın, kaza yapan birinci araca ait sürücünün cinsiyeti kadın ve erkek değerlerini alır. Kadın ve erkek kategorilerine doğrudan bir sayısal değere atanması daha sonra yapılacak işlemlerde anlamsızlığa sebep olur. Bu amaçla kategorik değere sahip öznitelikler için sayısallaştırma uygulanır. Şekil 3.5'te yapılan sayısallaştırma işlemi için bir örnek verilmiştir.

Sürücü Cinsiyeti		Sürücü Cinsiyeti_Kadın	Sürücü Cinsiyeti_Erkek
Kadın		1	0
Erkek		0	1
Erkek		0	1
Kadın		1	0
Erkek		0	1

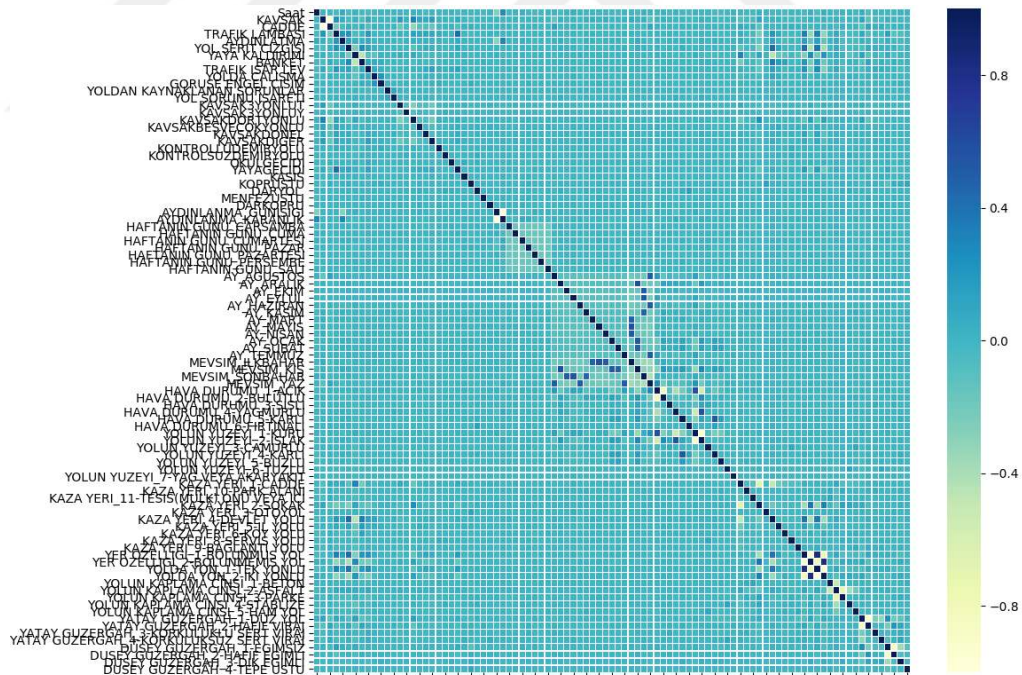
Şekil 3.5. Kategorik değıřkenlerin sayısallaştırılması

3.3.2. Temel istatistikler

Bu bölümde Eskişehir’de yaşanan trafik kazaları için temel istatistiksel yöntemlerle elde edilen sonuçlar değerlendirilmiştir. Önleme aşamasından sonra kazalar ile ilgili özniteliklerin aralarında ilişki bulunup bulunmadığı tespit etmek için korelasyon matrisleri oluşturulmuştur.

Korelasyon, iki değişken arasındaki doğrusal ilişkiyi gösterir. Korelasyon değeri -1 ile +1 değerleri arasında hesaplanır ve değişkenlerin arasında ilişki yoksa “0” değerini alır.

Şekil 3.6’da ölümlü-yaralanmalı kazalar için hesaplanan korelasyon matrisi görselleştirilmiştir. Oluşturulan korelasyon matrisi incelendiğinde birbiriyle ilişkili olduğu düşünülen öznitelikler dışında güçlü korelasyon bulunmamaktadır. Örneğin, hava durumu öznitelikleri ile yol yüzeyi öznitelikleri arasında, yol yönü ile yol bölümleri öznitelikleri arasında güçlü ilişkiler vardır. Bunun dışında zamansal öznitelikler arasında güçlü ilişkiler bulunmuştur.

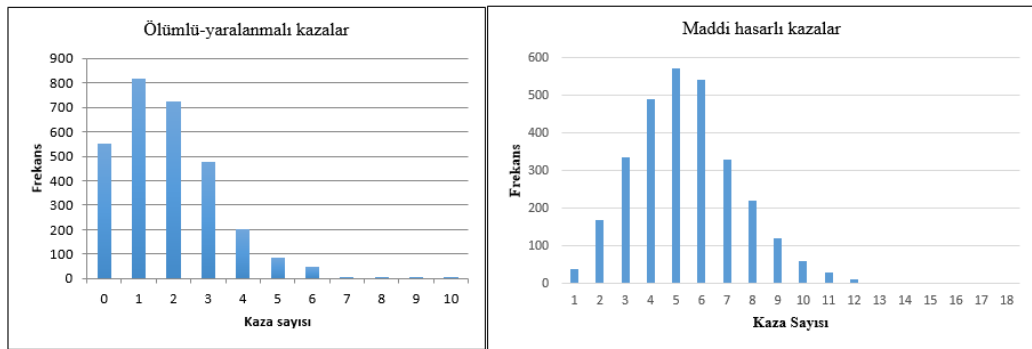


Şekil 3.6. *Ölümlü-yaralanmalı kazalara ait öz niteliklerin korelasyon matrisi*

3.4. Kaza Gerçekleşme Olasılıklarının Belirlenmesi

Trafik Kazaları için genellikle Poisson (Nicholson ve Wong, 1993) ve Negatif Binom olasılık dağılımları uygun görülmektedir. Bir verinin belirlenen bir dağılıma uyup uymadığını belirlemek için iyi uyum testi uygulanır. Kikare İyi uyum testi, veri ile veriye uyduğu varsayılan dağılım arasındaki farka göre dağılımın veriye uyup uymadığını gösteren istatistiksel bir testtir.

Trafik kazası olma olasılıklarının belirlenmesi için Eskişehir’de 2005-2012 yılları arasında sadece caddelerde gerçekleşen 2920 adet ölümlü-yaralanmalı ve adet maddi hasarlı kazalar incelenmiştir. Caddelerde ve kavşaklarda gerçekleşen ölümlü&yaralanmalı ve maddi hasarlı kaza olma olasılıklarını belirleyebilmek için frekans dağılımları incelenmiştir. Bu amaçla oluşturulan frekans dağılımlarına ait histogramlar Şekil 3.9’da verilmiştir.



Şekil 3.9. Kazaların frekans dağılımları

Trafik kazaları modellerinde çoğunlukla Poisson ve Negatif Binom dağılımları kullanılır (Lord ve Mannering, 2010), (Abdel-Aty ve Radwan, 2000). Her iki frekans dağılımına da bakıldığında kazaların olma olasılıklarının bahsedilen dağılımlara uygun olduğu söylenebilir.

Kazaların Poisson dağılımına uygun olduğu varsayımı altında dağılımların λ parametreleri tahmin edilmiştir ve dağılımların verileri temsil edip etmediğini test etmek için her iki tür kaza için de iyi uyum testleri uygulanmıştır.

X_i rastgele değişkeni 1 günde meydana gelen ortalama ölümlü&yaralanmalı kaza sayısını ifade etmektedir. Bu durumda ölümlü ve yaralanmalı kazaların Poisson dağılımına uygun olduğu varsayımı altında, Poisson dağılımı parametresi olan $\lambda_i=1.8281$

kaza/gün olarak tahmin edilmiştir. Dağılımın veriye uygun olup olmadığını test etmek üzere aşağıdaki hipotezler kurulmuştur:

Hipotezler	Test İstatistiği	Karar Kuralı
H_0 :Ölümlü&yaralanmalı kazalar parametresi 1.8281 kaza/gün olan Poisson dağılımına uymaktadır.	$\chi^2_0 = \frac{(O_i - E_i)^2}{E_i}$	$\chi^2_0 > \chi^2_{\alpha, k-p-1}$ ise H_0 red
H_a : Ölümlü&yaralanmalı kazalar parametresi 1.8281 kaza/gün olan Poisson dağılımına uymamaktadır.		

%95 güven seviyesinde $\chi^2_0 = 3.30594E-12 < \chi^2_{0.05, 25-1-1} = 15.507$ olduğundan H_0 reddedilemez. Ölümlü&yaralanmalı kazalar parametresi 1.8281 kaza/gün olan Poisson dağılımına uymaktadır.

Benzer şekilde X_2 rastgele değişkeni 1 günde meydana gelen ortalama maddi hasarlı kaza sayısını ifade etmektedir. Maddi hasarlı kazaların Poisson dağılımına uygun olduğu varsayımı altında, Poisson dağılımı parametresi olan $\lambda_2=4.4602$ kaza/gün olarak tahmin edilmiştir. Dağılımın veriye uygun olup olmadığını test etmek üzere aşağıdaki hipotezler kurulmuştur:

Hipotezler

Test İstatistiği

Karar Kuralı

H_0 : Maddi hasarlı kazalar parametresi 4.4602 kaza/gün olan Poisson dağılımına uymaktadır.

$(X_2 \sim \text{Poisson}(4.4602))$

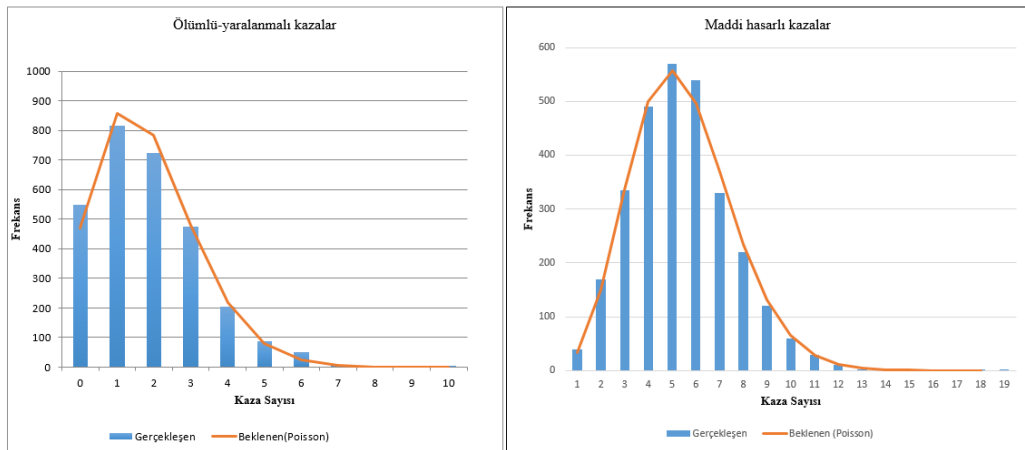
$$\chi_0^2 = \frac{(O_i - E_i)^2}{E_i}$$

$\chi_0^2 > \chi_{\alpha, k-p-1}^2$
ise H_0 red

H_a : Maddi hasarlı kazalar parametresi 4.4602 kaza/gün olan Poisson dağılımına uymamaktadır.

%95 güven seviyesinde $\chi_0^2 = 15.262 < \chi_{0.05, 25-1-1}^2 = 35.172$ olduğundan H_0 reddedilemez. Maddi hasarlı kazalar parametresi 4.4602 kaza/gün olan Poisson dağılımına uymaktadır.

Sonuç olarak Eskişehir’de bir günde gerçekleşen ölümlü-yaralanmalı ve maddi hasarlı kazaların Poisson dağılımına uygun olduğu istatistiksel olarak doğrulanmıştır. Belirlenen dağılımların verilere uygunluğu Şekil 3.10’dan da gözlemlenebilir:



Şekil 3.10. Belirlenen olasılık dağılımlarına göre beklenen ve gerçekleşen değerler

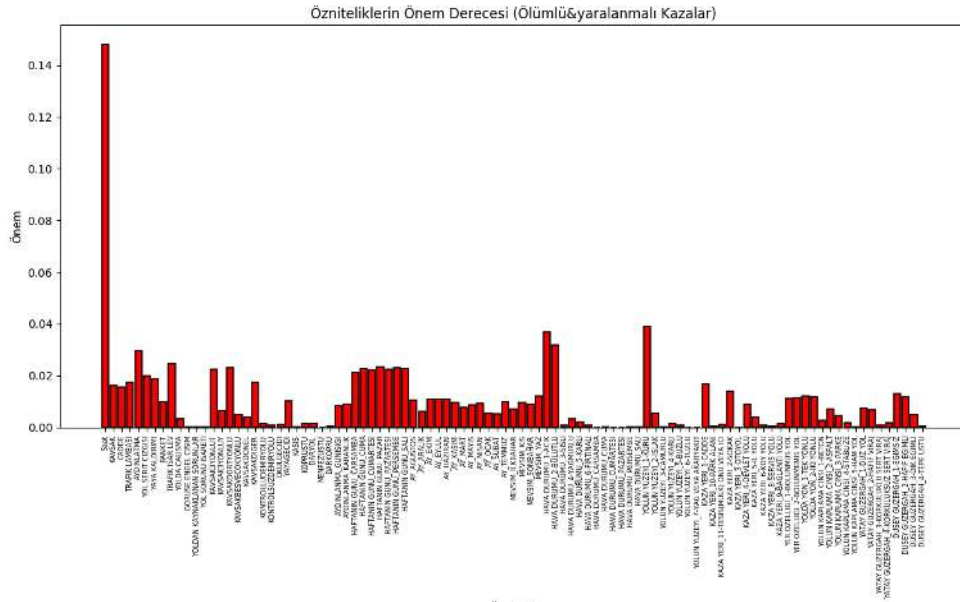
3.5. Öznitelik Seçimi

Öznitelik seçimi bir veristinde ilgilenilen çıktıya en çok katkıda bulunan özniteliklerin belirlenmesidir. Öznitelik seçimi ile veri setinin boyutu küçüldüğünden

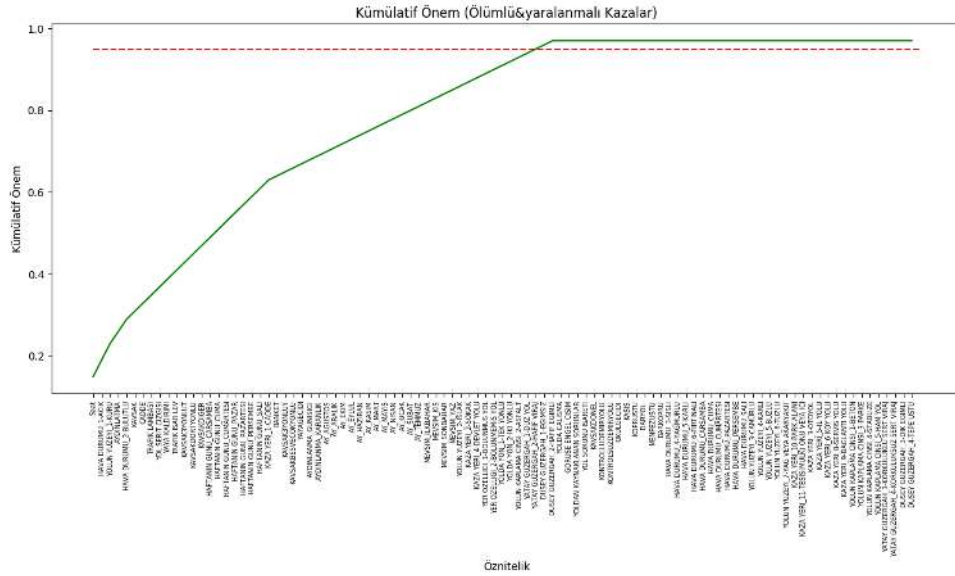
eğitim süreleri azalırken, aşırı öğrenme probleminin önüne geçildiği bilinmektedir. Önışleme aşamasından sonra uygulanan öznitelik seçimi ile veri seti sadece faydalı olarak görülen öznitelikleri içeren bir biçime indirgenir. Öznitelik seçimi yöntemleri filtre, sarmal ve gömülü yöntemler olmak üzere üç başlık altında incelenebilir. Filtre yöntemleri, bir önışleme süreci gibi bir ölçeğe göre belirlenen yüksek dereceli öznitelikleri seçer. Sarmal yöntemlerde öznitelik seçim kriteri, bir sınıflandırıcının performansdır, sınıflandırıcı en yüksek sınıflandırma performansını veren bir altkümeyi bulan bir arama algoritmasına sarılır. Gömülü yöntemler ise verileri eğitim ve test setlerine bölmeden eğitim sürecinin bir parçası olarak öznitelik seçimi içerir (Chandrashekar ve Sahin, 2014). Çalışmada öznitelik seçimi için kullanılan RO algoritması gömülü yöntemler içinde yer alır.

Kazalar için önemli öznitelikleri belirlemek amacıyla ayrı ayrı incelenen kazalar için RO ile özniteliklere ait önem dereceleri hesaplanmıştır. Hesaplanan önem dereceleri büyükten küçüğe sıralanarak birikimli önem dereceleri oluşturulmuştur. Böylece toplam önemin %95'ini oluşturan öznitelikler belirlenmiştir. Şekil 3.11 ve Şekil 3.12'de sırasıyla ölümlü-yaralanmalı kazalar için öznitelik önemleri ve büyükten küçüğe sıralanan birikimli önemler verilmiştir.

Şekil 3.11'den görüleceği üzere en yüksek önem derecesine sahip öznitelik kaza saatidir. Bunun dışında aydınlatma, trafik işaret levhası, haftanın günü, hava durumu, yol yüzeyi gibi bazı özniteliklerin de diğer özniteliklere oranla daha önemli olduğu görülmüştür.



Şekil 3.11. Ölümlü-yaralanmalı kazaların öznitelik önem dereceleri



Şekil 3.12. Ölümlü-yaralanmalı kazaların öznitelik birikimli önem dereceleri

Önişleme aşamasından sonra ölümlü-yaralanmalı kazalar versieti 91 adet öznitelğe sahiptir. Ölümlü-yaralanmalı kazalar veri seti öznitelik seçimi aşamasından sonra Şekil 3.12’de belirtildiği üzere toplam önemin %95’ini oluşturan 59 öznitelik seçilerek indirgenmiştir.

Benzer şekilde Şekil 3.13’te verilen maddi hasarlı kazalara ait önem derecelerine göre kaza saati aydınlatma, yol yüzeyi gibi öznitelikler yüksek önem derecesine sahipken;



Şekil 3.13. Maddi hasarlı kazaların öz nitelik önem dereceleri



Year	Cumulative Share (Oran)
1980	0.52
1985	0.58
1990	0.62
1995	0.68
2000	0.70
2005	0.72
2010	0.75

3.6. Eskişehir’de Gerçekleşen Kazaların Kümelenmesi

Kümeleme analizi olarak da bilinen veri kümelemenin amacı, bir dizi desen, nokta veya nesnenin doğal gruplandırmasını (unsurlarını) keşfetmektir (Jain, 2010). Kümeleme analizi için en çok kullanılan yöntemlerden biri K-ortalamlar algoritmasıdır.

K-ortalamlar kümeleme, bir veri kümesini k gruplarına otomatik olarak ayırmak için yaygın olarak kullanılan bir yöntemdir. “ k ” başlangıç küme merkezlerini seçerek ve ardından aşağıdaki adımları iteratif olarak uygulamaktadır.

1. Her örnek d_i en yakın küme merkezine atanır.
2. Her bir küme merkezi C_j , örneklerinin ortalaması olarak güncellenir.

Algoritma, örneklerin kümelerine atanmasında başka bir değişiklik olmadığında sona erer (Wagstaf vd., 2001). K-ortalamlar yöntemi büyük veri kümelerini kümelemede etkinliği için iyi bilinir (Huang, 1998).

K-ortalamlar yönteminin “ k ” küme sayısının belirlenmesinde literatürde çeşitli yöntemler önerilmiştir. Başparmak yöntemi, dirsek yöntemi, bilgi ölçüt yaklaşımı, gölge analizi, çapraz doğrulama bu yöntemlerden bazılarıdır. Çalışmada dirsek yöntemi kullanıldığından bu yöntemle ilgili bilgi verilmektedir.

Küme sayısı belirlemede en eski yöntemlerden biri olan dirsek yöntemin kümeleneceklerin sayısının bir fonksiyonu olarak açıklanan varyans yüzdesine bakan bir yöntemdir. Dirsek yönteminin adımları aşağıdaki gibi sıralanabilir:

1. k 'nın farklı değerleri için kümeleme algoritması uygulanır (Örneğin, k 1'den 10'a kadar değiştirilerek)
2. Her k için hata kareleri ortalama (SSE) değerleri hesaplanır.
3. SSE eğrisi çizilir.
4. SSE eğrisindeki dirsek (kırılma) noktası uygun küme sayısı olarak alınır.

Kümeleme analizi sonucu ölümlü-yaralanmalı kazalar için 6 adet, maddi hasarlı kazalar için 4 adet küme oluşturulmuş ve her iki tür kaza için kümelerde yer alan kaza karakteristikleri incelenmiştir.. Oluşturulan kümelerde yer alan kaza sayıları Tablo 3.2’de verilmiştir.

Tablo 3.2. Kümeleme analizi sonucu oluşturulan kümeler ve içerdikleri kaza sayıları

Kaza türü	Küme	Kaza sayısı
Ölümlü-yaralanmalı kazalar	1	320
	2	1325
	3	336
	4	991
	5	54
	6	395
Maddi hasarlı kazalar	1	1781
	2	64
	3	4093
	4	23028

Ölümlü-yaralanmalı kazalar için oluşturulan birinci kümede çoğunlukla bir ya da iki aracın karıştığı kazalar yer alır. Hava durumu açık ve bulutlu, yol yüzeyi ise kurudur. Kazaların çoğu caddede yandan çarpma çarpışma ve yayaya çarpma şeklinde oluşur. Kazaya sebep olan ilk araçlar %40 bisiklet %21 kamyonudur. Araçların %93'ü ticari amaçla kullanılmaktadır. Kaza sonucu 0-2 arası ölü, çok sayıda yaralı oluşmaktadır.

İkinci küme, çoğunlukla bir ya da iki aracın karıştığı kazaları içermektedir. Hava durumu açık, yol yüzeyi ise kurudur. Kazaların %61'i caddede, %36'sı sokakta gerçekleşir. Kazaya karışan ve çoğunlukla özel amaçlı kullanılan ilk araçların %56'sı otomobil, %16'sı kamyonettir. Kazaya karışan ilk aracın %24'ü iki tekerli araçlardır. Kaza sonucu 0-2 arası ölü, en fazla 10 yaralı oluşmaktadır.

Genellikle iki aracın karıştığı kazaların yer aldığı üçüncü kümede hava durumu açık, yol yüzeyi ise kurudur. Kazaya karışan ve çoğunlukla özel amaçlı kullanılan ilk araçların %62'si otomobil, %17'si kamyonettir. Kazaların %69'u caddede %25'i devlet yolunda gerçekleşir. Kaza sonucunda 0-4 arası ölü ve 0-9 arası yaralı ile sonuçlanmaktadır.

Dördüncü kümede çoğunlukla bir ya da iki aracın karıştığı kazalar yer alır. Hava durumu açık, yol yüzeyi kuru ve yollar çoğunlukla düz yol veya hafif virajlıdır. Yandan çarpma veya yandan çarpışma şeklindeki kazaları içeren bu kümedeki kazaların çoğu 0-3 ölü ve 0-11 arası yaralı ile sonuçlanmaktadır.

Tek araçlı kazaların yer aldığı beşinci kümede kazaların tümü korkuluksuz sert viraj olan cadde statülü yollarda gerçekleşir. Hava durumu açık, yol yüzeyi ise kurudur. Kazaya karışan birinci araç çoğunlukla otomobildir. Bu kazalar çoğunlukla 0-2 ölü veya 1-5 yaralı oluşmaktadır.

Oluşturulan son küme olan altıncı küme ise çoğunlukla bir ya da iki aracın karıştığı kazaları içermektedir. Hava durumu bulutlu ve yağmurlu, yol yüzeyi ise ıslaktır. %70 oranında caddelerde meydana gelen bu kazalarda 0-2 arası ölü ve 0-11 arası yaralıyla sonuçlanmaktadır.

Maddi hasarlı kazalar için oluşturulan dört kümeye ait özellikler incelendiğinde kümelerin genellikle kazaya karışan araçlara göre kümelendiği görülmüştür. Maddi hasarlı kazalar için oluşturulan birinci kümede Üç ve dört araçlı kazalar yer almaktadır ve asli kusurlu aracın cinsi %65 otomobildir. Bu kümede yer alan kazalardaki asli kusurlar çoğunlukla arkadan çarpma, doğrultu manevraları yanlış yapma, kavşakta geçiş önceliği, park halindeki araçlara çarpmadır. %46'sı orta hasarla sonuçlanan kazalarda, sürücülerin %52'sinin yaşı 18-35 arası, %40'ının yaşı 36-55 arasındadır.

Oluşturulan ikinci küme dört araçlı kazaları içermektedir. Bu kazalarda asli kusurlu aracın cinsi özel ve ticari amaçlarla kullanılan %58 otomobil ve %23 kamyonettir. %54'ü orta derece maddi hasarla sonuçlanan kazalarda sürücülerin %50'sinin yaşı 18-35 arası, %45'inin yaşı 36-55 arasındadır.

Maddi hasarlı kazaların üçüncü kümesinde çoğunlukla özel ve ticari amaçlarla kullanılan %70'i otomobil olan tek araçlı kazalar yer almaktadır ve kazaların %51'i hasarsız sonuçlanmıştır. Bu kazaların oluşmasına sebebiyet veren asli kusurlar çoğunlukla doğrultu ve manevraları yanlış yapma, kavşakta geçiş önceliği, manevra kurallarına uymama, park halindeki araçlara çarpma ve şerit ihlalleridir. Kazaya karışan birinci aracın sürücülerinin %56'sının yaşı 18-35 arası, %40'ının yaşı 36-55 arasındadır.

Dördüncü küme ise çoğunlukla özel ve ticari amaçlarla kullanılan %58'i otomobil %21'i kamyonet ve raylı araçları içeren iki araçlı kazalardır. Bu kazalarda asli kusurlar çoğunlukla arkadan çarpma, doğrultu ve manevraları yanlış yapma, kavşakta geçiş önceliği, hız kurallarına uymama ve şerit ihlalidir. %48'i düşük dereceli maddi hasarla sonuçlanan bu kazalarda da asli kusurlu aracın sürücülerin %47'sinin yaşı 18-35 arası, %44'ünün yaşı 36-55 arasındadır.

3.7. Eskişehir İçin Trafik Kaza Tahmini

Bölüm 3.1’de verilen literatür taramasından da görüleceği üzere, trafik kazaları ile ilgili değişkenler uzun yıllar makine öğrenmesi ve istatistiksel yöntemler kullanılarak tahmin edilmeye çalışılmıştır. Ayrıca, tüm bu çalışmalarda gerçek kaza verileri dikkate alınmıştır. Literatürden farklı olarak, bu çalışmada kaza tahmin problemi tek sınıflı bir sınıflandırma problemi olarak tanımlanmıştır. Birçok çalışmaya benzer şekilde, uygulanan tek sınıf sınıflandırma modelinin performansını değerlendirmek üzere elde edilen sonuçlar ikili sınıflandırma modelleri ile karşılaştırılmıştır.

3.7.1. Ters örnek oluşturma

Trafik kaza verisetleri çoğunlukla sadece kaza olan durumları içerir. Bu amaçla trafik kaza tahmin problemi bir tek sınıf sınıflandırma problemi olarak tariflenmiştir. Tek sınıf sınıflandırma modelini test etmek amacıyla kaza veri seti kaza olmayan durumları içerecek şekilde genişletilmiştir. Kaza olmayan durumları veri setine eklemek için zaman ile ilgili öznitelikler seçilmiştir. Bunun sebebi veri setinde üretilebilecek yeni verilerin mekânsal özniteliklerinin Eskişehir’i temsil etmemesi riskidir. Mekânsal özniteliklerinin değiştirilmesi halinde veri setinde Eskişehir’de hiç yer almayan türde yol ve yapılar içermesinin önüne geçilmiştir. Ters örnek oluşturma prosedüründe seçilen öznitelikler ve değişim aralıkları Tablo 3.3’te verilmiştir.

Tablo 3.3. Değişebilir öznitelikler

Öznitelik	Değer aralığı
Saat	[0,23]
Gün	{Pazartesi, Salı, Çarşamba, Perşembe, Cuma, Cumartesi, Pazar}
Mevsim	{İlkbahar, yaz, sonbahar, kış}
Yol yüzeyi	{Kuru, ıslak, çamurlu, karlı, buzlu, tuzlu, yağ veya akaryakıt dökülmüş}

Dengeli bir veri seti oluşturmak için, oluşturulan kazasız örneklerin sayısı varolan kaza verisinin büyüklüğü kadardır. Uygulanan ters örnek oluşturma prosedürü Tablo 3.4’te verilmiştir.

Tablo 3.4. Ters örnek oluşturma prosedürü

Girdi	<i>Kaza veri seti, değişebilir öznitelikler listesi, örnek sayısı</i>
Çıktı	<i>Veri seti</i>
1:	<i>Başla</i>
2	<i>Veri seti = [], Üretilen örnek sayısı = 0</i>
3:	<i>Üretilen örnek sayısı < örnek sayısı:</i>
4:	<i>Kaza veri setinden rasgele bir örnek seç</i>
5:	<i>Değişebilir öznitelikler listesinden rasgele bir öznitelik seç</i>
6:	<i>Seçilen özneliğin değerini değer kümesinden rastgele bir değer seçerek değiştir ve Yeni Satır'a ata.</i>
7:	<i>Eğer Yeni satır Veri setinde yoksa:</i>
8:	<i>Yeni satırın sınıf değerini "0" olarak ayarla</i>
9:	<i>Yeni satırı Veri setine ekle</i>
10:	<i>Üretilen örnek sayısı +=1</i>
11:	<i>Adım 3'e git.</i>
12:	<i>Eğer Yeni satır Veri setinde varsa:</i>
12:	<i>Adım 3'e git.</i>
13:	<i>Bitir</i>

Saat ve mevsim ile ilgili öznitelikler kullanılarak oluşturulan ters örneklerde sırasıyla aydınlanma ve ay öznitelikleri de eş zamanlı olarak saat veya mevsime uygun biçimde değiştirilmiştir.

3.7.2. Hesaplamalı sonuçlar

Tek sınıf sınıflandırma modelinin başarısını karşılaştırmak amacıyla ikili sınıflandırma modelleri oluşturulmuştur. K-en yakın komşu, destek vektör regresyon, rasgele orman sınıflandırıcıları ikili sınıflandırma modellerini oluşturmak için seçilmişlerdir.

İkili sınıflandırma modellerinin parametrelerinin seçimi için grid search yöntemi kullanılmıştır. Her bir sınıflandırıcı için kullanılan parametre aralıkları Tablo 3.5'te verilmiştir.

Tablo 3.5. *Model parametre aralıkları*

Sınıflandırıcı	Parametre
kNN	k: [1,10]
DVM	Kernel: {Doğrusal, R} C: {1, 10, 100, 1000} γ : {0.001, 0.0001}
RO	$mtry$: $\{\sqrt{\text{öznitelik sayısı}}, \log_2(\text{öznitelik sayısı}), \text{öznitelik sayısı}\}$ $ntree$: {100, 300, 500, 800, 1000}
ODVM	Kernel: {Doğrusal, RBF, Polinom} Aykırı değer oranı: {0.25, 0.50, 0.75, 0.95}

Kaza olmayan durumlara ait veriler oluşturulduktan sonra meydana gelen verisetlerinin büyüklükleri Tablo 3.6’da verilmiştir. Parantez içinde ise verisetlerinin içinde yer alan sırasıyla kaza olan ve olmayan örnek sayıları verilmiştir.

Tablo 3.6. *Veri seti büyüklükleri*

Sınıflandırıcı	Veri seti	
	Ölümlü-yaralanmalı	Maddi hasarlı
Tek sınıf	8280 (8280, -)	16337 (16377, -)
İki sınıf	16560 (8280, 8280)	32754 (16377, -)

Bahsedilen sınıflandırıcılar hem tüm veri seti için hem de RO sınıflandırıcı ile elde edilen indirgenmiş veri seti için test edilmiştir. Her iki durum için de eğitim ve test veri seti büyüklükleri aynıdır.

Kesinlik ve duyarlılık metriklerinin harmonik ortalaması olan F-skoru, sınıflandırıcı performanslarının ölçümü için kullanışmıtır. F-skoru, tek sınıf sınıflandırma probleminde uygun bir performans değerlendirme ölçütüdür (Sokolova, Japkowicz ve Szpahowicz, 2006). Kesinlik, duyarlılık ve F-skoru Denklem (3.10), (3.11) ve (3.12) kullanarak hesaplanır:

$$Kesinlik = \frac{TP}{TP + FP} \quad (3.10)$$

$$Duyarlılık = \frac{TP}{TP + FN} \quad (3.11)$$

$$F - \text{skoru} = 2 \times \frac{\text{Kesinlik} \times \text{Duyarlılık}}{(\text{Kesinlik} + \text{Duyarlılık})} \quad (3.12)$$

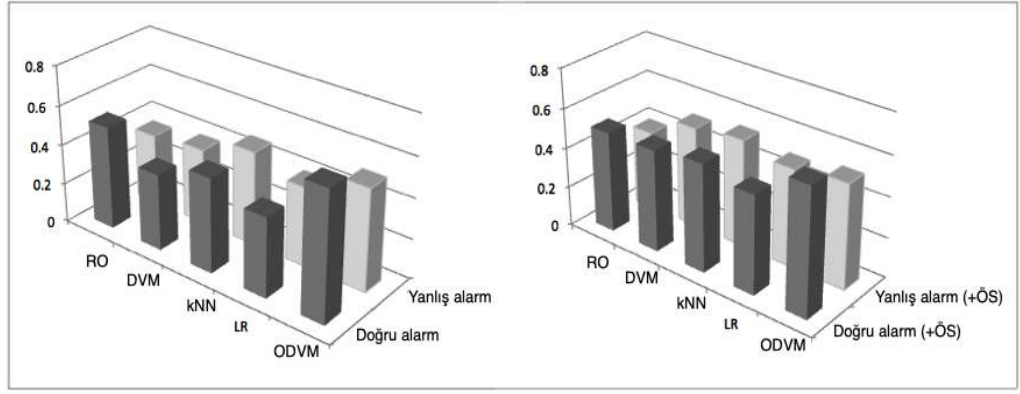
TP, TN, FP ve FN sırasıyla sınıflandırıcı ile elde edilen doğru pozitif, doğru negatif, yanlış pozitif ve yanlış negatif tahmin sayılarını ifade eder.

10 katlamalı çapraz doğrulama ile yapılan testler sonucunda ölümlü-yaralanmalı ve maddi hasarlı kaza verisetleri için elde edilen F-skorları Tablo 3.7’de verilmiştir. ODVM ile diğer sınıflandırıcılara göre daha iyi sonuçlar elde edildiği söylenebilir.

Tablo 3.7. Testler sonucu elde edilen F-skorları

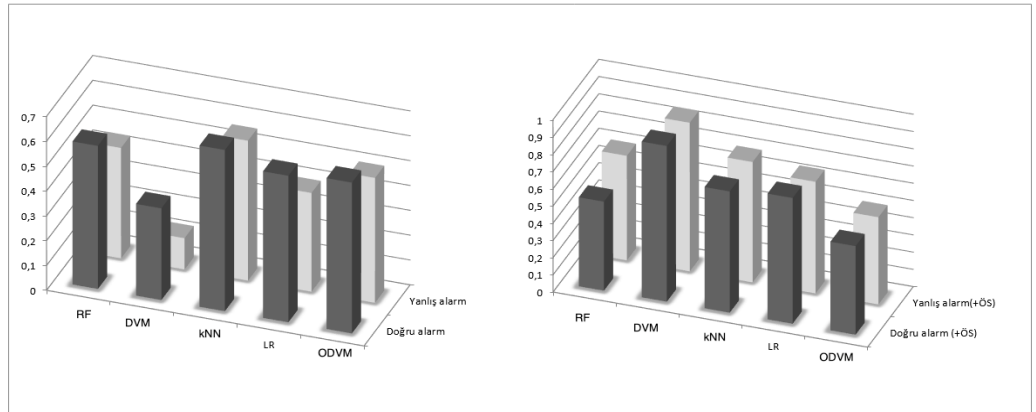
Sınıflandırıcı	Ölümlü-yaralanmalı		Maddi hasarlı	
	F skoru	F skoru (indirgenmiş)	F skoru	F skoru (indirgenmiş)
kNN	0.486	0.519	0.612	0.615
LR	0.519	0.564	0.640	0.643
RO	0.552	0.531	0.591	0.510
DVM	0.252	0.582	0.501	0.688
O-DVM	0.600	0.588	0.600	0.672

F-skoruna ek olarak, doğru ve yanlış kaza alarm oranları da hesaplanmıştır. Yanlış alarm oranı, yanlışlıkla kaza olarak tahmin edilen normal durumların oranını ifade eder. (Lin, Wang ve Sadek, 2015), doğru alarm oranı ise doğru tahmin edilen kaza oranını belirtmektedir. Şekil 3.15’te ölümlü-yaralanmalı kazalar için doğru ve yanlış alarm oranları verilmiştir. ODVM sınıflandırıcısı hem doğru hem de yanlış alarm oranlarında en yüksek değere sahiptir. Test edilen diğer iki sınıf sınıflandırıcılarının doğru alarm oranlarının düşük olduğu görülmektedir. Bu da test edilen sınıflandırıcıların gerçek bir kaza durumunu gözden kaçırabileceğini göstermiştir.



Şekil 3.15. Ölümlü-yaralanmalı kazalar için doğru ve yanlış alarm oranları

Maddi hasarlı kazalar için yapılan tahminler sonucu hesaplanan doğru ve yanlış alarm oranları Şekil 3.16’da verilmiştir. Tüm kaza very seti için DVM en düşük yanlış alarm oranına sahip olsa da, doğru kaza alarm oranı oldukça düşüktür. ODVM ile elde edilen doğru alarm oranı DVM dışındaki diğer iki sınıf sınıflandırıcılarla rekabet edebilir olduğu görülmüştür. İndirgenmiş very setinde için hesaplanan doğru ve yanlış alarm oranlarına göre ODVM ile en düşük yanlış ve alarm oranlarına ulaşılmıştır.



Şekil 3.16. Maddi hasarlı kazalar için doğru ve yanlış alarm oranları

3.8. Sonuç

Bu bölümde Eskişehir’in trafik güvenliğinin geliştirilmesi amacıyla geçmişte gerçekleşen trafik kazaları üzerinde çalışılmıştır. Eskişehir’de gerçekleşen kazalara ait veriler düzenlenerek kazalarla ilgili tanımlayıcı analitikler sunulmuş; kazaların gerçekleşme olasılık dağılımları belirlenmiştir. Maddi hasarlı ve ölümlü yaralanmalı

kazalar ayrı ayrı ele alındığında bir günde gerçekleşen ortalama kaza sayılarının Poisson dağılımına uygun olduğu iyi uyum testiyle doğrulanmıştır. Ek olarak, kaza verisetlerinde yer alan özniteliklerin önem dereceleri RO yöntemi ile belirlenmiştir.

Kaza gerçekleşmesini tahmin edebilmek için farklı modeller üzerinde çalışılmıştır. Literatürde daha çok iki sınıf sınıflandırma problemi olarak ele alınan kaza tahminleme problemi, bu çalışmada kazalar için daha önce örneğine rastlanılmamış olan tek sınıf sınıflandırma problemi olarak ele alınmıştır. Ölümlü kazalar için oluşturulan tek sınıf sınıflandırma modeli ile başarılı sonuçlar elde edilirken; maddi hasarlı kazaları tahmin etmede tek sınıf sınıflandırma modeli, karşılaştırılan iki sınıf sınıflandırma modelleri ile rekabetçi sonuçlar elde edilebilmiştir. Kaza tahminlemesi için tek sınıf sınıflandırma modelinin performansını geliştirmek amacıyla kaza olmayan durumlara ait veriler toplanarak verisetlerine eklenmelidir. Ayrıca farklı tek sınıf sınıflandırma modelleri oluşturularak performansı yükseltmek amacıyla topluluk yaklaşımları uygulanabilir.

4. DİNAMİK VE STOKASTİK TRAFİK AĞLARINDA ROTA ÖNERİSİ

Trafik ağları yapısı gereği belirsiz ve dinamiktir. Trafik durumu sürekli değişen ve değişen koşullardan etkilenen bir yapıya sahiptir. Trafik sıkışıklığı, trafik kazaları, yol yapımı vb. sebepler trafik durumunu doğrudan etkiler. Trafikte geçirilen süre trafik durumuna bağlı olarak değişir. Bu sebeple, belirli bir çıkış noktasından bir varış noktasına geçen süreyi ifade eden seyahat süresi belirsizdir. Trafikteki bu belirsizlikler ve trafik koşullarının dinamik olması sebebiyle, istenilen varış noktasına mesafe olarak en kısa olan rota, çoğunlukla süre olarak en kısa rota olmayabilir. Trafikte seyahat süresi hesaplanırken trafiğin durumunu göz önüne almak en kısa süren rotanın bulunmasını sağlayabilir. Aynı zamanda trafikte araçların geçirdiği sürenin azalması uzun dönemde araçların atmosfere salmış olduğu karbon emisyonunu da büyük ölçüde azaltacaktır.

Trafikte rota bulma mekanizmaları çevrimiçi ya da çevrimdışı olarak çalışır. Çevrimiçi rota bulma mekanizmaları, çoğunlukla GPS, sokak kamerası veya sensör türü anlık veri toplayan cihazları kullanabilir. Çevrimiçi öneri yapabilmek için rota önceden planlanmıştır; araçlar rotada ilerlemeye başladığında, algoritma değişkenlerin değişikliklerini izleyecek ve yeniden rotalama ihtiyacını kontrol edecektir (İsa, Mohamed, & Yusoff, 2015). Çevrimdışı rota bulma mekanizması ise seyahat başlamadan rota oluşturur; stokastik seyahat süreleri ve tahminler ile çalışabilir.

Bu çalışmada trafikteki belirsizlikleri göz önüne alan bir en kısa yol problemi üzerinde çalışılmıştır. Problem çevrimdışı olarak ele alınmış; stokastik seyahat süreleri ile daha önceki bölümlerde bahsedilen tahmin modelleri bir kuralcı analitik uygulaması olan çözüm yaklaşımına eklenmiştir. Başka bir deyişle; geçmiş verilerden elde edilen bilgiler doğrultusunda sürücü yola çıkmadan önce bir rota belirlemek amacıyla bir yaklaşım uygulanmıştır. Kazaların ve trafik akışının yarattığı belirsizlik sebebiyle bağlantı yollarındaki seyahat süresinin de belirsiz olduğu varsayımı altında bir sonlu Markov Karar Süreci modeli oluşturulmuştur. Problemin stokastik kısmı için kaza olasılık dağılımları ve trafik akış olasılıkları hesaplanmıştır. Oluşturulan model bir dinamik programlama yaklaşımı ile geriye doğru değer yinelemesi algoritması ile çözülmüştür. Markov Karar Sürecinin çözümü için önerilen geriye doğru değer yinelemesi algoritması ile elde edilen çözüm, deterministik problemlerde kullanılan Dijkstra algoritması ile elde edilen çözüm ile karşılaştırılmıştır. Karşılaştırmalar, oluşturulan test problemleri ve farklı türdeki senaryolar üzerinden yapılmıştır.

4.1. Literatür Taraması

Trafikte rota bulma problemi, literatürde en kısa yol problemi, rota bulma, rota önerisi vb. başlıklar altında incelenmiştir. Bu konudaki çalışmalar genellikle trafiği farklı yapılarda ele almıştır.

Dener ve diğerleri (2011), problemi zamana bağlı ve dinamik olarak ele almış ve Ankara için oluşturmuş oldukları örnek problem için GA temelli bir çözüm yöntemi önermişlerdir. Qi ve Schneider (2015), trafikteki seyahat süresini artıran durumları EKS problemine yansıtmak amacıyla, rotanın toplam seyahat süresine sağa dönüş oranı, rotadaki trafik ışığı sayısı ve trafiği etkileyecek diğer durumları ifade eden bir değişkeni belirlemiş oldukları çeşitli katsayılarla çarparak amaç fonksiyonuna eklemişlerdir. Oluşturdukları problemin çözümü için ilk aşaması “Graph Partitioning” olan bir rota bulma sezgiseli önermişlerdir. Sun ve diğerleri (2017), trafikte rota belirleme için trafik ağının esnekliğini ve hiyerarşisini göz önüne alarak, sabit bir zamanda başlamak ve hedefe olabildiğince erken ulaşmak ve yoğun saatlerden kaçınmak için en iyi yola çıkış zamanını seçmek amaçlarıyla iki adet Bellman-Ford algoritması temelli sezgiseller önermişlerdir. Taksilerden toplanan veriler aracılığıyla yapmış oldukları testlerde Bellman-Ford, A* arama ve bazı probleme özel geliştirilen algoritmalarından daha iyi sonuç elde ettiklerini raporlamışlardır. Hu ve diğerleri ise (Hu, ve diğerleri, 2018), A* arama algoritması ile birleştirdikleri enerji tüketim modelinde trafik ışığına yakalanma olasılıkları ile araçların yavaşlama ve hızlanma durumlarını da göz önüne almışlardır. Sever ve diğerleri (2018), geçmiş ve gerçek zamanlı verileri kullanarak dinamik ve zaman bağımlı stokastik bozulmaları içeren EKS problemi üzerine çalışmışlardır. Geçmiş veri aracılığıyla ağdaki düğümler arasındaki seyahat süresi stokastik olarak ele alınırken; gerçek zamanlı veri aracılığıyla ağ dinamik hale getirilmiştir. Problem kesikli zamanlı sonlu Markov Karar Süreci olarak ele alınmış ve kümeleme ile yaklaşık Dinamik Programlama algoritması ile çözülmüştür.

Literatürde yapılan çalışmalarda da görüldüğü üzere trafik ağında EKS probleminin çözümü için genellikle probleme zaman boyutu, stokastik seyahat süreleri kullanılmış; trafik ağı dinamik olarak ele alınmaya çalışılmıştır. Bu sebeple, yapılan çalışmada Sever ve diğerlerinin (2018) yapmış olduğu çalışma baz olarak alınmış; trafik ağı geçmiş verilerden yola çıkarak elde edilen olasılık dağılımları ve daha önceki bölümlerde elde edilmiş tahminlerden faydalanılarak stokastik ve dinamik olarak ele alınmıştır.

4.2. En Kısa Yol Problemleri

V düğümler ve A ayrıtlar arasındaki ayrıtlar olmak üzere verilen bir $G = (V, A)$ yönlü ağı ve her $(i, j) \in A$ için ayrıt maliyeti c_{ij} için, en kısa yol problemi başlangıç düğümü s ve bitiş düğümü t arasındaki en az maliyetli yolu bulmaya çalışır. Bir en kısa yol probleminde bir düğümün birden fazla ziyaret edilmediği varsayımı olduğunda edilen rota temel olarak adlandırılır.

En kısa yol problemleri, tek kaynaklı ve tüm ikililer arasında en kısa yol problemleri olmak üzere iki türe sahiptir. Tek kaynaklı en kısa yol problemi bir düğümden diğer tüm düğümlere olan en kısa yolu bulmayı amaçlarken; tüm ikililer arasında en kısa yol problemi tüm düğüm çiftleri arasındaki en kısa yolu belirlemeye çalışır.

Verilen G ağında $\delta^+(i)$ ve $\delta^-(i)$ sırasıyla bir i düğümüne gelen ve i düğümünden giden ayrıtları ifade eder. (i, j) ayrıtının maliyeti c_{ij} olmak üzere, bir s düğümünden bir t düğümüne en kısa yolu belirlemek için tamsayılı model kullanılabilir (Taccari, 2016).

Karar değişkeni:

$$x_{ij} = \begin{cases} 1, & (i, j) \text{ ayrıtı rotadaysa} \\ 0, & \text{diğer durumlarda} \end{cases}$$

EKS problemi için bir standart bir tamsayılı model aşağıdaki gibi verilebilir.

$$\min \sum_{(i,j) \in A} c_{ij} x_{ij} \quad (4.1)$$

$$\sum_{(i,j) \in \delta^+(i)} x_{ij} - \sum_{(i,j) \in \delta^-(i)} x_{ji} = \begin{cases} 1 & i = s \text{ ise} \\ -1 & i = t \text{ ise} \\ 0 & \text{diğer durumlarda} \end{cases} \quad \forall i \in V \quad (4.2)$$

$$\sum_{(i,j) \in \delta^+(i)} x_{ij} \leq 1 \quad \forall i \in V \quad (4.3)$$

$$x_{ij} \in \{0,1\} \quad \forall (i,j) \in A \quad (4.4)$$

Problemin amacı minimum seyahat maliyeti Denklem (4.1)'de verilmiştir. Denklem (4.2) akış kısıtıdır. Denklem (4.3), her bir düğümden en fazla 1 çıkışın olmasını sağlar. Denklem (4.4) karar değişkeni için işaret kısıtıdır. Ayrıt maliyetlerinin negatif olması durumunda modele alt tur engelleme kısıtları eklenmelidir.

Tek kaynaklı EKS probleminin çözümünde Dijkstra algoritması, Bellman-Ford algoritması, A* arama algoritması vb. algoritmalar kullanılır.

Dijkstra algoritması, bir ağdaki düğümler arasındaki en kısa yolu bulmak için bilgisayar mühendisi Edsger W. Dijkstra (1959) tarafından geliştirilmiştir. Tüm ayrıt maliyetlerinin pozitif olması ve yönlü bir ağ olması durumunda, Dijkstra algoritması ağdaki bir başlangıç düğümünden diğer tür düğümlere olan en kısa yolu bulabilir. Verilen G ağında başlangıç noktası s düğümünden diğer düğümlere olan en kısa yol maliyetleri Dijkstra algoritması ile Tablo 4.1’de verildiği gibi hesaplanır (Reedy, 2013).

Tablo 4.1. *Dijkstra algoritması*

Girdi	V ağı, s başlangıç noktası, w uzaklık matrisi
Çıktı	$Uzaklık$
1:	<i>Başla</i>
2:	$Uzaklık(v) = 0$
3:	<i>Her $v \in V(s)$ için:</i>
4:	$Uzaklık(v) = \infty$
5:	$S = \{ \}$
6:	$Q = V$
7:	$Q = \{ \}$ olana kadar:
8:	$u = \min Uzaklık(Q, Uzaklık)$
9:	$S = S \cup \{u\}$
10:	<i>Her $v \in Komşu(u)$ için:</i>
11:	<i>Eğer $Uzaklık(v) > Uzaklık(u) + w(u, w)$:</i>
12:	$Uzaklık(v) = Uzaklık(u) + w(u, w)$
13:	<i>Bitir</i>

Dijkstra algoritmasına benzer olan Bellman-Ford algoritması negatif ağ maliyetleri söz konusu olduğunda kullanılır. Negatif maliyetler rotada döngülere sebep olabilir ve Dijkstra algoritması bu durumun önüne geçemez. Dijkstra algoritması her iterasyonda bulunulan düğüme bağlı olan düğümlerle ilgilenirken; Bellman-Ford algoritması her iterasyonda tüm ayrıtları inceler.

EKS probleminin çözümünde kullanılan bir başka algoritma da Dijkstra algoritmasının bir uzantısı olan A* algoritmasıdır. Hart, Nilsson ve Raphael (1968) tarafından önerilen A* ("yıldız" olarak), yol bulmada yaygın olarak kullanılan, düğümler

olarak adlandırılan noktalar arasında verimli bir şekilde yönlendirilebilir bir yol çizme işlevi olan bir bilgisayar algoritmasıdır (Reedy, 2013).

EKS’de ele alınan ağ genellikle statiktir. Buna karşılık, bir ağ dinamik olabilir, düğümler ve ayrıtlar zaman içinde eklenebilir, güncellenebilir ya da silinebilir (Madkour vd., 2017). Dinamik ağlar ağın kenarlar, köşeler veya ağırlıklar gibi farklı parametrelerinin güncellenmesi açısından hiçbir kısıtlamasının olmadığı problemlerdir (Sunita ve Garg, 2018). Dinamik ağların zaman boyutu vardır. Her bir zaman dilimi için düğümler ağa eklenir ya da ağdan çıkarılır; ayrıt maliyetleri güncellenir.

4.3. Markov Karar Süreci

Sonlu durumlu, sonlu eylemli Markov karar süreci belirsizlik altında, özellikle basit ve nispeten izlenebilir bir sıralı karar verme modelidir (White, 2011). Bir Markov Karar Süreci (MKS) modeli şunları içerir (Givan ve Parr, 2001):

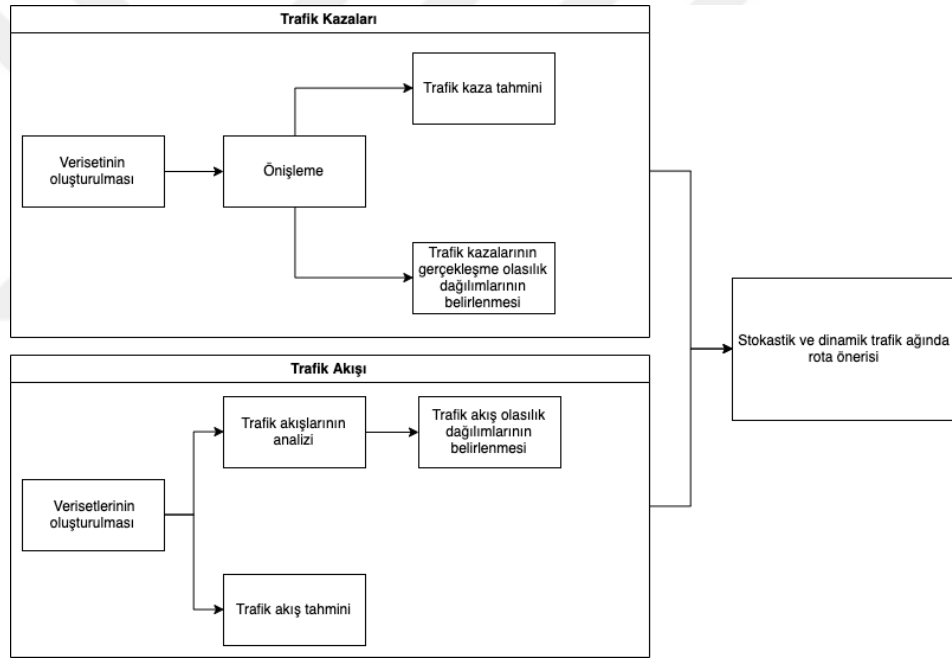
- Olası durumlar kümesi S
- Olası hareket kümesi A
- Gerçek bir değerli ödül fonksiyonu (s, a)
- Her bir eylemin her bir durumdaki etkisi T

MKS, Markov özelliğini taşır. Yani bir $t+1$ anında sistemin durumu S , yalnızca istemin t anındaki S' durumuna bağlıdır. Sonlu olmayan MKS’lerin çözümünde değer yinelemesi, politika yinelemesi ve doğrusal programlama kullanılır. Değer yinelemesinin sonucu Bellman denkleminin çözümü olan tek bir rota verir. Buna karşın politika yinelemesi ise başlangıçta rastgele seçilmiş bir politikadan en iyi durağan politikayı verir. Doğrusal programlama ile politika yinelemesi gibi optimal durağan politika elde edilir. Sonlu MKS’lerin çözümü için dinamik programlama kullanılabilir.

MKS’nin geçiş olasılıkları ve ödülleri belirlendikten sonra değer yinelemesi yöntemi uygulanır. Değer yinelemesi yöntemi geriye doğru dinamik programlamaya benzer (Song vd., 2000). Değer yinelemesi algoritması Bellman tarafından geliştirilen iteratif bir süreçtir. Algoritma her durumun beklenen faydasını komşu durumların faydalarını kullanarak ve birbirini izleyen durumlar yeterli derecede birbirine yaklaşıncaya kadar hesaplamaya devam eder.

4.4. Dinamik ve Stokastik Ağlarda En Kısa Yol Problemi için Dinamik Programlama Temelli Bir Çözüm Yaklaşımı

Problemin çözümü için geliştirilen Markov Karar Süreci temelli çözüm yaklaşımında sürücünün bir başlangıç noktasından bitiş noktasına gideceği rota, sürücü yola çıkmadan önce belirlenmektedir. Problem çevrimdışı olarak ele alınmıştır, yani sürücü yolda iken herhangi bir anlık veri sisteme gelmemekte ve rota yola çıkmadan belirlenmektedir. Sistemle ilgili tüm veriler geçmiş verilerden elde edilen olasılık dağılımları ve tahminler aracılığıyla elde edilmektedir. Kazalar ve trafik akışı için önceki bölümlerde bahsedilen tahmin modellerinin çıktılarının tamamen doğru olduğu varsayımı ile ele alınan trafik ağında, düğümlerin ağdaki varlıkları her 15 dakikalık zaman dilimi için güncellenmektedir. Çözüm yaklaşımının kapsamı Şekil 4.1’de özetlenmiştir.



Şekil 4.1. Çalışmanın kapsamı

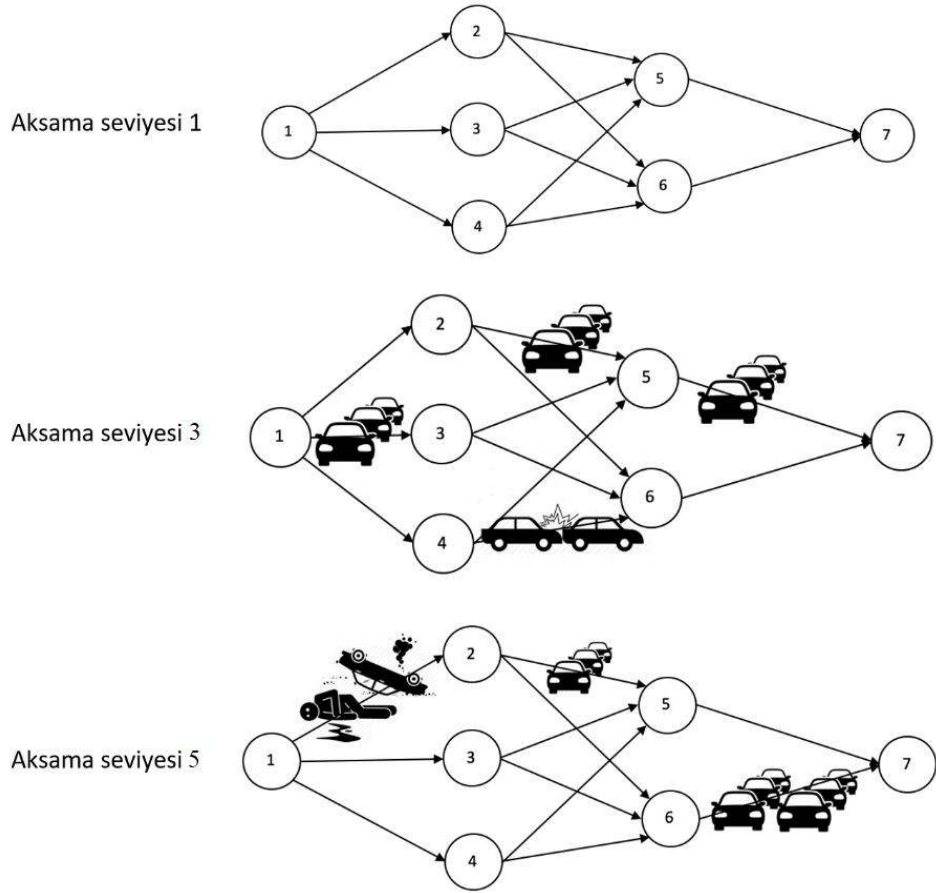
Problem tek kaynaklı bir EKS problemi olarak düşünülmüştür. Ele alınan problemde trafik ağı yönlü bir ağıdır ve karar verme noktaları kavşaklardır. Başka bir deyişle, kavşaklar trafik ağının düğümlerini, kavşaklar arasındaki yollar ise trafik ağının ayrıtlarını ifade etmektedir.

Trafiğin durumunun 15 dakikada bir güncellendiği varsayıldığından olağan trafik akışının aksamada beş tür durum olduğu varsayılmıştır. Bu durumlar trafikte yaşanan

aksamaların seviyesine bağlıdır. Birinci durumda herhangi bir aksamamanın yaşanmadığı, üçüncü durumda orta dereceli aksamaların ve beşinci durumda ise çok yüksek dereceli aksamaların yaşandığı varsayılmıştır. Aradaki seviyeler ise bir önceki aksama seviyesinin benzeri olup, trafik ışığı ya da seyahatten meydana gelen sürenin eklenmesi sebebiyle 15 dakikalık zaman dilimi aşımı için söz konusudur. Trafikte yaşanan her bir aksama zaman kaybına sebep olmaktadır. Bahsedilen durumlar Tablo 4.2’de özetlenmiş ve Şekil 4.2’de aksama seviyeleri için örnek verilmiştir.

Tablo 4.2. *Trafiğin aksama seviyeleri*

Aksama seviyeleri	Aksamaya sebep olan durumlar		Kaybedilen zaman (dakika)
	Trafik akışı	Kaza	
1: Aksama yok	Hafif	-	0
2: Düşük aksama	Hafif	-	15
3: Orta aksama	Orta dereceli trafik akışı	Maddi hasarlı	30
4: Yüksek aksama	-	-	45
5: Çok yüksek aksama	Ağır trafik akışı	Ölümlü-yaralanmalı	60



Şekil 4.2. Aksama seviyelerinde trafik ağının durumu

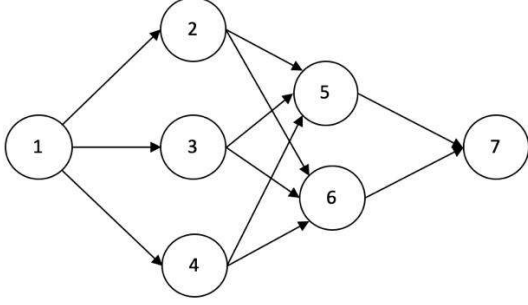
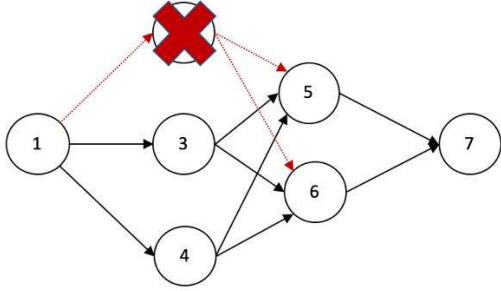
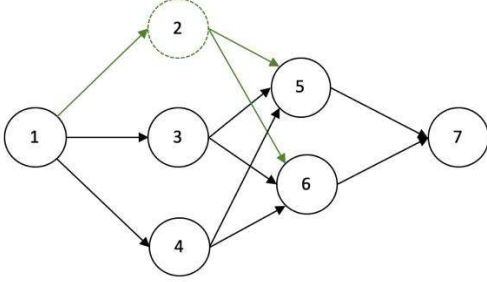
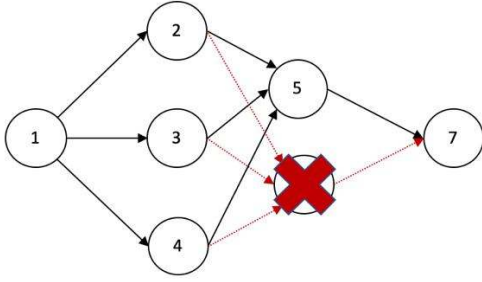
Trafikte aksamaya sebep olan durumların tamamen birbirinden bağımsız olduğu yaklaşımı altında kaza gerçekleşme olasılıkları ile trafik akışları ile ilgili olasılık değerleri geçmiş veriler aracılığıyla hesaplanmıştır. Bu şekilde elde edilen olasılık değerleri kullanılarak iki kavşak arasındaki beklenen seyahat süreleri hesaplanmaktadır.

Trafikte zaman kaybına sebep olan bir başka durum ise trafik ışıklarıdır. Trafik ışıklarında kırmızıya yakalanma durumunda yavaşlama ve kalkış anında hızlanma sebepleri ile bekleme süresine ek olarak zaman kaybı yaşanmakta; ayrıca duruş-kalkış sebebiyle yakıt tüketimi ve dolayısıyla maliyet ile emisyon artmaktadır. Bahsedilen maliyetleri ve zaman kayıplarını azaltmak amacıyla geliştirilen yaklaşımda trafik ışığına yakalanma durumu da göz önüne alınmıştır.

Dinamik ađlarda en kısa yol problemleri genellikle çevrimiçi modellerdir. Çalışmada çevrimiçi bilgi alınamadıđı için, trafik ađının dinamik hale getirilmesinde ise tahmin modelleri kullanılmıřtır. 15 dakikalık sabit zaman periyotları için yapılan tahminler sonucunda ölümlü ve yaralanmalı bir kaza gerçekleşeceđi ya da trafik akışının yoğun olacađı kavşak ya da yollar ađdan çıkarılır. Benzer şekilde daha önce ađdan çıkarılan bir kavşak ya da yolda tahmin sonucunda bir aksaklık tespit edilmezse ađa tekrar eklenir. Tablo 4.3'te bir ađa ait bir saatlik zaman dilimi için düğüm ekleme ve silme sürecine örnek verilmiřtir.



Tablo 4.3. *Düğüm ekleme ve silme işlemleri*

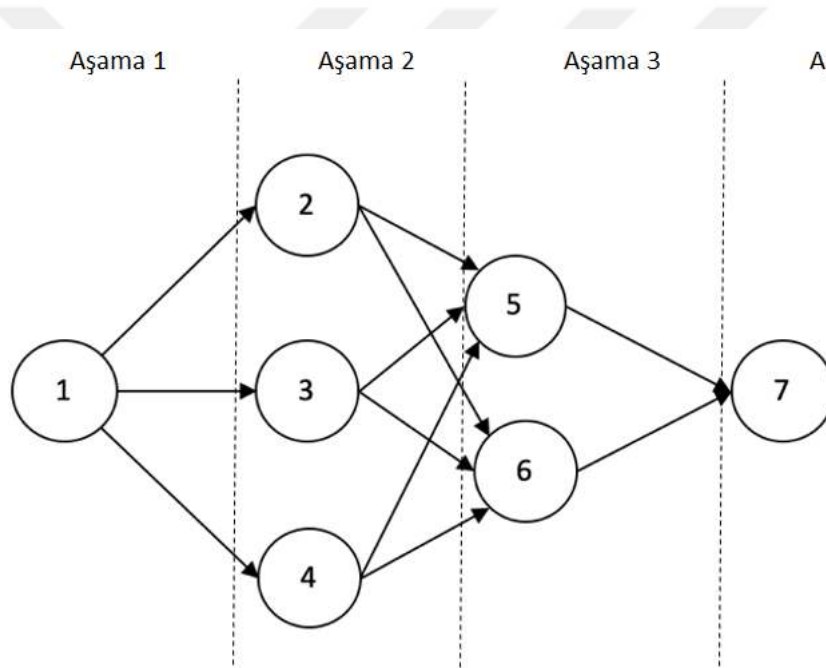
Zaman Aralığı	Örnek	İşlem
12:00:00-12:14:59		-
12:15:00-12:29:59		Düğüm silme
12:30:00-12:44:59		Düğüm ekleme
12:45:00-12:59:59		Düğüm silme

Belirtilen problemi çözmek üzere Markov Karar Süreci temelli geriye doğru dinamik programlama algoritması önerilmiştir. Önerilen çözüm yaklaşımı ile elde edilen rotaların değerlendirilmesi için sürücünün belirtilen başlangıç ve bitiş noktaları arasını belirlenen rota ile gerçekte ne kadar sürede kat ettiği hesaplanmıştır. Farklı seviyedeki

aksamalar için oluşturulan senaryolar üzerinden MKS temelli çözüm yaklaşımı ile elde edilen rotalar ve Dijkstra algoritması ile elde edilen rotaların gerçekleşme süreleri hesaplanmıştır ve karşılaştırılmıştır.

4.4.1. Durum uzayı

Bir k kararında, sistemin durumu S_k 'dir. Sistem durumunun 15 dakikada bir değişeceği varsayımı altında en kötü senaryoda bir kavşaktan diğer kavşağa gidiş süresi, sabit seyahat süresi ve en yüksek seviyede bir aksama gerçekleştiğinde ortaya çıkacak maliyet olan 60 dakikanın toplamıdır. Bu sebeple sistemin tüm durumları gözden geçirilebilmesi için her bir kavşağa ait en kötü senaryoda varılacak süreler göz önüne alınmıştır. Bahsedilen yaklaşım Şekil 4.3'teki örnek üzerinden anlatılacaktır.



Şekil 4.3. Bir ağ örneği

1 kavşağından 7 kavşağına en kısa yolu bulmak üzere sistem 4 aşamada ele alınır. Birinci aşamada 1 kavşağından 2,3 ve 4 kavşaklarına, ikinci aşamada 2,3 ve 4 kavşaklarından 5 ve 6 kavşaklarına, üçüncü aşamada ise 5 ve 6 kavşağından 7 kavşağına varılır. Bu örnek için en kötü senaryoda 7 kavşağına, kavşaklar arası deterministik seyahat süresine ek olarak 180 dakikada ulaşılabilir. Sistemin her 15 dakikada bir güncellendiği varsayıldığından en kötü durumda sistemin ne kadar güncelleneceği hesaplanır. Örneğin, 7 kavşağı için $180/15 = 12$ durum söz konusu olacaktır. Sistemde herhangi bir aksama

olmaması durumu da eklenerek toplamda 7 kavşağı için 13 durum geçerlidir. Benzer şekilde birinci aşamadaki kavşaklar için 5 durum, ikinci aşamadaki kavşaklar için 9 durum söz konusu olacaktır.

Sistemin herhangi bir aşamadaki bir durumunun hesaplanabilmesi için sistemin bir sonraki aşamaya ait ve bulunulan zamandan itibaren 60 dakikalık zaman diliminde yer alan durumlar kullanılır.

4.4.2. Aksiyon uzayı

Sistem S_k durumunda iken; bir i_k kavşağında verilecek karar seyahat edilecek bir sonraki x_k komşu kavşağıdır. Karar verilecek kavşak ile aralarında yol bulunan kavşaklar hareket edilebilecek komşu kavşaklar olarak adlandırılır.

4.4.3. Amaç fonksiyonu

Ele alınan problemin amaç fonksiyonu belirlenen bir başlangıç ve bitiş noktaları arasındaki seyahat süresinin enküçüklenmesidir. Kavşaklar arasındaki seyahat süresi sabit seyahat süresine ek olarak trafikteki belirsizliklerden dolayı gerçekleşen ilave sürenin toplamıdır. Araç hızının ($V=30$ km/sa) sabit olduğu varsayımı altında i ve j kavşakları arasındaki l_{ij} uzunluğundaki bağlantı yolunun sabit seyahat süresi (t_{ij}) Denklem (4.5)'te verildiği şekilde hesaplanır.

$$t_{ij} = \frac{l_{ij}}{V} \quad (4.5)$$

Trafikte gerçekleşen trafik sıkışıklığı ve trafik kazası sebepleri ile trafikte yaşanan aksamalardan kaynaklanan zaman kayıpları ise amaç fonksiyonun stokastik kısmını oluşturur.

İki kavşak arasında yer alan trafik ışıkları da trafikte geçirilen süreye etki eder. Bu amaçla kavşaklar arasındaki yollarda bulunan trafik ışıkları da amaç fonksiyonuna eklenmiştir. Trafik ışıklarının birbirinden bağımsız olduğu varsayımı altında bir trafik ışığına yakalanma olasılığı 0.5 olarak alınmış ve trafik ışığında kırmızıya yakalanma durumunda duruş, bekleme ve kalkış dahil ortalama 2 dakika kaybedilmesi belirlenmiştir. İki kavşak arasında trafik ışığına yakalanma olayı K ile ifade edilmiştir.

Problemin amaç fonksiyonu minimum beklenen seyahat süresidir. Seyahat süresi deterministik seyahat süresi ile trafik durumlarından dolayı meydana gelecek gecikmenin beklenen değerinin toplamı olarak ifade edilir. Denklem (4.6)'da sistemin her bir durumu

için, sistem S_k durumunda iken x_k kavşağına gitmenin maliyetlerinin toplamının enküçüğü olarak ifade edilir.

$$z = \min \sum_k c(S_k, x_k) \quad (4.6)$$

Sistemin her bir i_k karar noktasındaki maliyet Denklem (4.7) ile hesaplanır. Her bir i_k karar noktasının maliyeti her bir komşu karar noktası için hesaplanır. Hesaplanan her bir maliyet değeri karar ve komşu düğümler arasındaki deterministik ve trafik ışığı maliyetleri ile komşu kavşağın beklenen değerinin toplamı olarak ifade edilir.

$$v(i_k, x_k, S_k) = t_{i_k x_k}(S_k) + E[K_{i_k x_k}] + \sum_{S_{k-1}} P(S_{(k-1)}|S_k) V(x_k, S_{k-1}) \quad (4.7)$$

$t_{i_k x_k}(S_k)$, sistem S_k durumunda iken i_k ve x_k kavşakları arasındaki seyahat süresidir. Bu süre iki kavşak arasındaki sabit süre ile durumlar arası geçişten kaynaklanan sürenin toplamıdır.

Her karar noktasında her bir komşu kavşağa hareket edilmesi durumunda karşılaşılabilecek maliyetler hesaplandıktan sonra bu maliyetler arasındaki her bir durum en küçük değere sahip x_k kavşağa hareket edilir ve Denklem (4.8)'de verildiği şekilde i_k kavşağının maliyeti hesaplanan enküçük maliyet değerini alır.

$$V(i_k, S_k) = \min_{x_k \in Komşu(i_k)} v(i_k, x_k, S_k) \quad (4.8)$$

$Komşu(i_k)$ 'da yer alan kavşaklar her 15 dakikada bir sistemde elde edilen kaza ve trafik akış tahminlerine bağlı olarak güncellenir. Başka bir deyişle, her 15 dakikada bir $Komşu(i_k)$ kümesine düğüm eklenebilir ya da kümeden düğüm çıkarılabilir.

Sonuç olarak başlangıç kavşağı "0" olarak işaretlenirse Denklem (4.9) problemin çözümünü verecektir.

$$V_0 = V(0, S_0) \quad (4.9)$$

4.4.4. Geçiş matrisi

Geçiş matrisleri her bir yol için ayrı ayrı oluşturulmuştur. Yollara ait geçiş matrislerinin oluşturulması için daha önce bahsedilen aksama seviyeleri kullanılmıştır.

Geçiş matrisi, bir yol için üç durumu ve aralarındaki geçiş olasılıklarını içerir. Geçiş matrislerinin oluşturulması için aşağıdaki varsayımlar yapılmıştır:

- Aksama seviyesi 1 ve 2 dışındaki tüm seviyelerindeki aksamaların koşullu olasılık değerlerini belirlemek için kaza gerçekleşme ve trafik akış düzeylerinin birbirinden bağımsız olarak ele alınmıştır.
- Aksama seviyesi 1 ve 2 dışındaki tüm seviyelerindeki aksama seviyelerine ait durumlar arasında geçiş yoktur. Başka bir deyişle, bir zaman aralığında belirli bir yolda maddi hasarlı bir kaza varsa; bir sonraki zaman aralığında ölümlü-yaralanmalı kaza gerçekleşmez.
- Aksama seviyesi 3, 4 ve 5'in 1 ve 2 engelleme seviyesine geçiş olasılıkları 0.5 olarak varsayılmıştır. Yani, bir zaman aralığında belirli bir yolda gerçekleşen maddi hasarlı bir kazanın bir sonraki zaman aralığında ortadan kalkması olasılığı 0.5'tir.

Her bir i yolu için sistemin 5 aksama seviyesini içeren sistemin S_k durumundaki geçiş matrisi Denklem (4.10)'da verilmiştir.

$$\Theta_{S_k} = \begin{bmatrix} p_{11}^i & p_{12}^i & p_{13}^i & p_{14}^i & p_{15}^i \\ p_{21}^i & p_{22}^i & p_{23}^i & p_{24}^i & p_{25}^i \\ p_{31}^i & p_{32}^i & p_{33}^i & p_{34}^i & p_{35}^i \\ p_{41}^i & p_{42}^i & p_{43}^i & p_{44}^i & p_{45}^i \\ p_{51}^i & p_{52}^i & p_{53}^i & p_{54}^i & p_{55}^i \end{bmatrix} \quad (4.10)$$

Matriste her bir p^i değeri, i kavşağının bir aksama durumdan bir başka duruma geçmesi olasılığını ifade eder. Geçiş matrisi sistemde her 15 dakikada bir güncellenir. Böylece geçen süreye göre kaç adım ileriye gidileceği belirlenmiş olur. Hesaplanan seyahat süresi 15 dakikadan az ise matris aynı kalırken, 15 ile 30 dakika arasında ise matrise 1 adımlık geçiş uygulanır.

Daha önce kazalar için belirlenen olasılık dağılımlarına ait rastgele değişkenler ve parametre 1 gün içindir. Uygulanan çözüm yöntemi 15 dakikalık periyotları içerdiğinden hesaplanan λ parametreleri 15 dakikaya uygun biçimde güncellenmiştir.

4.4.5. Çözüm yaklaşımı

Dinamik ve stokastik ağlarda en kısa yol probleminin çözümü için önerilen geriye doğru dinamik programlama temelli algoritma Tablo 4.4'te verilmiştir.

Tablo 4.4. Çözüm yaklaşımı adımları

Girdi	<i>Kavşak kümesi (I), kavşak sayısı, çıkış (0), varış</i>
Çıktı	<i>Rota, minBeklenenSüre</i>
1:	Başla
2:	$kararSayısı = kavşak\ sayısı$
3:	$rota = []$
4:	$V(i_{varış}, S_k) = 0$
5:	Eğer $kararSayısı > 0$:
6	$\forall i_k \in I$:
6:	$v(i_k, x_k, S_k) = t_{i_k x_k}(S_k) + E[K_{i_k x_k}] + \sum_{S_{k-1}} P(S_{(k-1)} S_k) V(x_k, S_{k-1})$
7:	$V(i_k, S_k) = \min_{x_k \in Komşu(i_k)} v(i_k, x_k, S_k)$
10:	Her $x_k \in Komşu(i_k)$ için:
11:	Eğer $V(i_k, S_k) = V(x_k, S_k)$ ise:
12:	$rota_{i_k S_k} = i_k + rota_{x_k S_k}$
13:	$kararSayısı = kararSayısı - 1$
14:	$minBeklenenSüre = \min V(0, S_0)$
15:	Bitir

4.4.6. Senaryolar

Elde edilen rotaları değerlendirmek için trafikte gerçekleşebilecek durumları içeren 3 tür senaryo üretilmiştir. Birinci tip senaryo, trafiğin doğal akışını bozacak herhangi bir durumun gerçekleşmediği senaryodur. Bu senaryoda sadece trafik ışığının etkisi söz konusudur. İkinci tip senaryo trafiğin hafif etkilendiği durumları içerir. Bu durumlar bir yolun ya da kavşağın, çok kısa süreli ve genellikle kısmen trafiğe kapanmasına sebep olacak maddi hasarlı kazalar ile ortalama trafik akışının söz konusu olmasıdır. Son senaryo olan üçüncü tip senaryoda ise trafik akışında yoğun aksama durumlarının modele yansıtılması sağlanmaya çalışılmıştır. Ölümlü ve yaralanmalı kazalar ile çok yoğun trafik akışı bu senaryoda yer almaktadır. Ayrıca yoğun aksamaya ek olarak, ikinci tip senaryoda yer alan durumlar da üçüncü tip senaryoya dahildir.

Bahsedilen senaryolar her test problemi için ayrı ayrı olmak üzere 10'ar adet üretilmiştir. Ele alınan ağın rasgele olarak seçilen yollarında 3 ve 5 seviyesinde aksamalar oluşturulmuştur. Ayrıca her bir senaryo tipi için etkilenecek yol sayısı, ağdaki toplam yolların %10, %30, %50 ve %70'i olarak belirlenmiştir. Oluşturulan senaryolar Tablo

4.5’te özetlenmektedir. Böylece her bir test problemi için elde edilen rotalar, 9 tip senaryo üzerinden 10’ar farklı senaryo için değerlendirilmiştir.

Tablo 4.5. Senaryo bilgileri

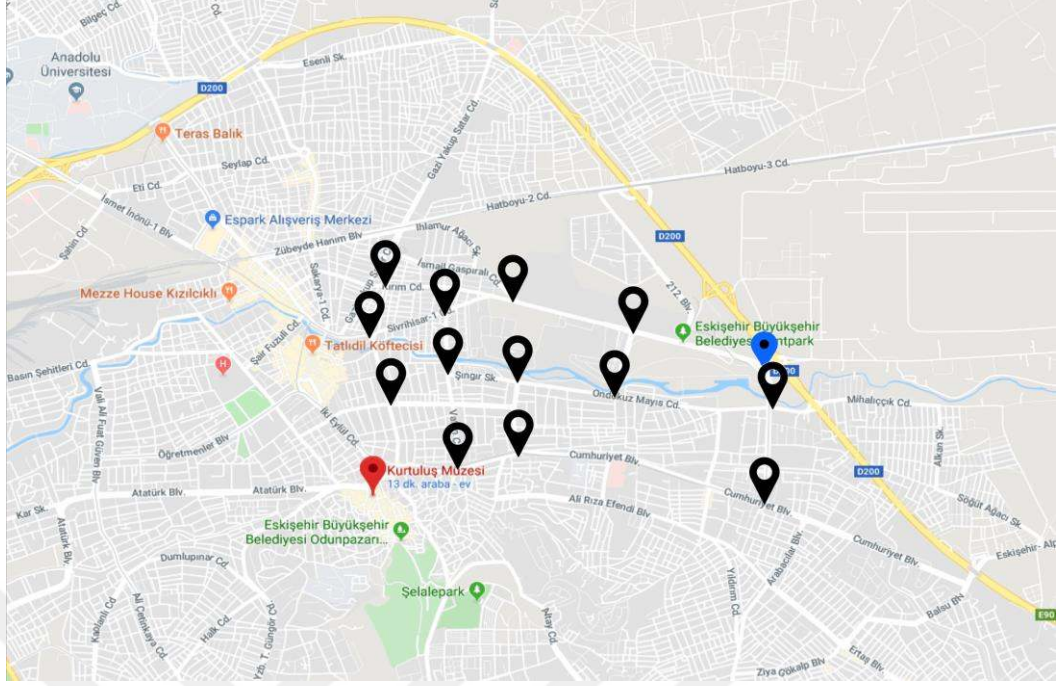
Senaryo	Etkilenen en fazla yol oranı	Sebep olan durumlar
1	-	-
2	%10 %30 %50 %70	Maddi hasarlı kazalar Ortalama trafik akışı
3	%10 %30 %50 %70	Ölümlü ve yaralanmalı kazalar Yoğun trafik akışı

Senaryolara ek olarak, amaç fonksiyonu hesaplamasında bir yol geçmek için seçildiyse tüm kırmızı ışıklara yakalanıldığı varsayılmıştır. Bunun amacı amaç fonksiyonuna yansıtılan trafik ışığı bölümünün etkinliğinin ölçülmesidir.

4.4.7. Test problemleri

Uygulanan çözüm yaklaşımını test etmek amacıyla 8 adet test problemi oluşturulmuştur. Bunlardan biri Eskişehir’den toplanan veriler aracılığıyla oluşturulmuştur.

Eskişehir test problemi, başlangıç ve bitiş düğümleri dahil 15 kavşaktan oluşmaktadır. Şekil 4.4’te problem için ele alınan ağ verilmiştir. Belirlenecek rotanın başlangıç noktası olarak Otogar kavşağı, bitiş noktası olarak ise Kurtuluş Müzesi civarındaki Yediler kavşağı seçilmiştir.



Şekil 4.4. Eskişehir test problemi

Eskişehir test probleminde trafik ağının ele alınan kısmı için trafik ışığı sayıları oluşturulmuş ve yapılan tahminler ile ağ dinamikleştirilmiştir. Eskişehir test problemi için oluşturulan senaryonlardan biri ise Tablo 4.6’da verilmiştir.

Tablo 4.6. Eskişehir test problem için oluşturulan senaryolara örnekler

Senaryo	Zaman aralığı	Gerçekleşen durum
Senaryo2	16:00-16:15	Yunus Emre Caddesi 1. kısım Orta derecede trafik akışı
		Yunus Emre Caddesi 2. kısım Orta derecede trafik akışı
	16:15-16:30	Yunus Emre Caddesi 1. kısım orta derece trafik akışı
	16:30-16:45	Cumhuriyet Bulvarı 2. kısım maddi hasarlı kaza
	16:45-17:00	Cumhuriyet Bulvarı 1. kısım orta derece trafik akışı
Senaryo3		Cumhuriyet Bulvarı 2. kısım orta derece trafik akışı
	16:00-16:15	-
	16:15-16:30	Cumhuriyet Bulvarı 3. kısım ölümlü&yaralanmalı kaza
		Sivrihisar-2 Caddesi ağır derecede trafik akışı
	16:30-16:45	Sivrihisar-2 Caddesi orta derecede trafik akışı
	16:45-17:00	Sarper Bulvarı ağır derecede trafik akışı
		Cumhuriyet Bulvarı 3. kısım ağır derecede trafik akışı

Eskişehir test problemine ek olarak farklı boyutlarda test problemleri oluşturulmuştur. Düğüm sayısı 10, 30, 50, 100, 200, 500 ve 1000 olmak üzere oluşturulan test problemleri için ayrıtlardaki trafik ışıkları, zamana bağlı olarak ağdaki düğüm bilgileri, Senaryo 2 ve Senaryo 3'ün her bir durumu için etkilenecek olan ayrıtlar rastgele olarak oluşturulmuştur.

4.4.8. Hesaplamalı sonuçlar

Önerilen çözüm yaklaşımı Intel(R) Core™ i7 işlemcili 2.5 GHz ve 16GB RAM'e sahip işlemcili bir bilgisayarda Python 3.5.4 versiyonu ile kodlanmıştır. Dijkstra algoritması ise Python Networkx paketi aracılığı ile çalıştırılmıştır.

Elde edilen rotaların değerlendirilmesi için Bölüm 4.4.6'da belirtilen senaryolar için amaç fonksiyonu değerleri hesaplanmıştır. Eskişehir test problemi için elde edilen rotanın 9 tip senaryo için elde edilen sonuçlara ait ortalamalar Tablo 4.7'de verilmiştir.

Tablo 4.7. Eskişehir örnek problemi için elde edilen sonuçlar

Senaryo	Etkilenen en fazla yol oranı (%)	Dijkstra	MKS	Fark (%)
Senaryo 1	-	3.280	3.280	-
Senaryo 2	10	3.280	3.280	-
	30	3.280	3.280	-
	50	6.280	6.280	-
	70	15.280	15.280	-
Senaryo 3	10	3.280	3.280	-
	30	12.280	12.280	-
	50	36.280	36.280	-
	70	24.280	24.280	-

Tablo 4.2'den de görüldüğü üzere küçük boyutlu Eskişehir test problemi için her iki yaklaşımla aynı rotalar elde edilmiş ve tüm senaryolar değerlendirildiğinde sonuçlar arasında herhangi bir fark olmadığı görülmüştür.

Senaryo 1 için test problemlerine ait sonuçların ortalamaları Tablo 4.8'de verilmiştir.

Tablo 4.8. *Senaryo 1 için test problemlerine ait elde edilen ortalama seyahat süreleri (dakika)*

Düğüm sayısı	Dijkstra (dakika)	MKS (dakika)	Fark (%)
10	1.004	1.004	-
30	9.461	1.651	-82.55
50	9.479	1.899	-79.97
100	27.437	4.307	-84.30
200	50.2	49.533	-1.33
500	241.486	194.452	-19.48
1000	182	126	-30.77

Özellikle daha büyük boyutlu problemlerde kaza ve sıkışıklık gibi bir aksamanın olmadığı durumda başarılı sonuçlar elde edilmiştir. Çözümlerde deterministik yaklaşımla trafik ışığına yakalanma durumu göz önüne alınmadığı için daha kısa seyahat süreleri elde edilebilmiştir.

Senaryo 2 için test problemlerine ait sonuçların ortalamaları Tablo 4.9’da verilmiştir.

Tablo 4.9. Senaryo 2 için test problemlerine ait elde edilen ortalama seyahat süreleri (dakika)

Düğüm sayısı	Etkilenen en fazla yol oranı					
	%10			%30		
	Dijkstra	MKS	Fark (%)	Dijkstra	MKS	Fark (%)
10	1.004	1.004	-	1.004	1.004	-
30	12.461	4.651	-62.67	9.461	4.651	-50.84
50	9.479	1.899	-79.97	12.479	4.899	-60.74
100	27.437	7.307	-73.37	30.437	13.307	-56.28
200	50.2	49.533	-1.33	53.2	49.533	-6.89
500	241.486	194.452	-19.48	241.486	197.452	-18.24
1000	182	126	-30.77	182	126	-30.77

Düğüm sayısı	Etkilenen en fazla yol oranı					
	%50			%70		
	Dijkstra	MKS	Fark (%)	Dijkstra	MKS	Fark (%)
10	1.004	1.004	-	4,008	4,008	0
30	9,461	1,651	-82,55	18,461	13,651	-26,05
50	9,479	7,899	-16,67	18,479	13,897	-24,80
100	30,437	4,307	-85,85	33,437	7,307	-78,15
200	56,2	49,533	-11,86	50,2	52,533	4,65
500	241,486	197,452	-18,24	244,486	197,452	-19,24
1000	185	132	-28,65	182	129	-29,12

Senaryo 2 için elde edilen sonuçlar incelendiğinde genellikle uygulanan yaklaşımla daha kısa seyahat süreleri elde edilebilmiştir. Bu da ağın durumundaki aksamaların artmasının sonucudur.

Senaryo 3 için test problemlerine ait sonuçların ortalamaları Tablo 4.10’da verilmiştir.

Tablo 4.10. Senaryo 3 için test problemlerine ait elde edilen ortalama seyahat süreleri (dakika)

Düğüm sayısı	Etkilenen en fazla yol oranı					
	%10			%30		
	Dijkstra	MKS	Fark	Dijkstra	MKS	Fark
10	1.004	1.004	-	13,004	13.004	-
30	12.461	4.651	-62.67	15,479	7.651	-50.57
50	9.479	1.899	-79.97	15,479	7.899	-48.97
100	27.437	4.307	-84.30	57,467	7.307	-87.28
200	53.2	52.533	-1.25	62,2	52.533	-15.54
500	241.486	194.452	-19.48	241,486	197.452	-18.24
1000	182	126	-30.77	182	132	-27.47

Düğüm sayısı	Etkilenen en fazla yol oranı					
	%50			%70		
	Dijkstra	MKS	Fark	Dijkstra	MKS	Fark
10	7.004	7.004	-	16.004	1.004	-
30	12.461	10.651	-14.52	27.479	28.651	4.27
50	9.479	1.899	-79.97	21.479	22.899	6.61
100	33.467	4.307	-87.13	48.435	4.307	-91.11
200	62.2	63.675	2.37	68.2	76.533	12.22
500	253.486	200.452	-20.92	241.486	177.552	-26.48
1000	188	126	-32.98	197	129	-34.52

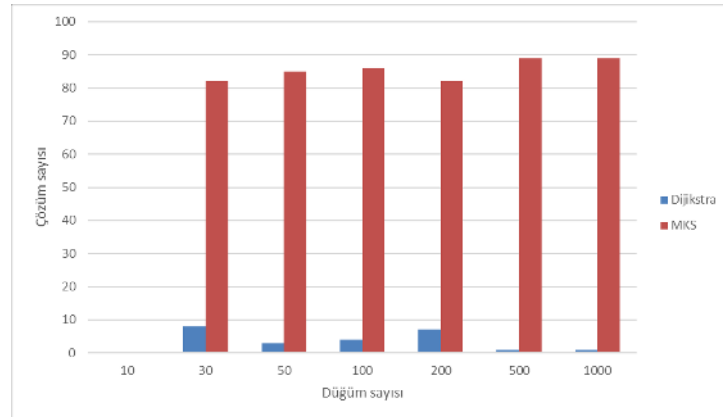
En ağır aksamaların yaşandığı Senaryo 3 için elde seyahat süreleri incelendiğinde önerilen MKS temelli yaklaşım ile deterministik çözüm yöntemine baskın çözümler elde edilebilmiştir. Senaryo 2 ve Senaryo 3 birlikte ele alındığında, aksama seviyeleri ve etkilenen yol oranları arttıkça MKS temelli yaklaşım daha kısa seyahat süreleri sağlayabilmektedir. Senaryolarda oluşturulan ağlarda rastgele aksamalar oluşturulduğundan elde edilen sonuçların senaryo bazındaki ortalamaları ise Tablo 4.11’de verilmiştir.

Tablo 4.11. Elde edilen ortalama ortalama seyahat süreleri (dakika)

Düğüm sayısı	Dijkstra	MKS	Fark (%)
10	6.004	6.004	-
30	14.132	8.651	-38.78
50	12.812	7.232	-43.55
100	35.11	6.307	-82.04
200	56.2	55.105	-1.95
500	243.153	194,574	-19.98
1000	184.667	128	-30.69

Tablo 4.10'dan da görüleceği üzere problemin amaç fonksiyonuna trafik ışıklarının, kaza ve trafik akış olasılıklarının eklenmesi sonuçları olumlu yönde etkilemiştir. Böylece deterministik çözüm yaklaşımları ile ele alamadığımız gerçek trafik koşulları da ele alınmıştır.

Test problemleri için her bir senaryo türü için oluşturulan 10'ar adet trafik simülasyonu için elde edilen sonuçlar ayrı ayrı incelenmiş ve önerilen MKS yaklaşımının baskın bir şekilde daha kısa seyahat süreleri sağladığı görülmüştür. Uygulanan tüm testlerde MKS ve Dijkstra'nın baskın gelme sayıları Şekil 4.5'te görselleştirilmiştir.



Şekil 4.5. Uygulanan testlerde çözüm yöntemlerinin baskın gelme sayıları

4.5. Sonuç

Trafik ağında rota önermek için trafiğin değişen ve belirsiz koşullarını göz önüne almak gerekir. Bu amaç doğrultusunda, çalışmada trafik ağında rota önermek için geçmiş trafik verilerinden elde edilen olasılık dağılımları ve tahminler aracılığıyla ağıdaki

belirsizlikler ve deęiřebilecek durumlar en kısa yol hesaplamasına dahil edilmiřtir. Bir kuralcı analitik uygulaması önerilen MKS temelli çözüm yaklaşımı çevrimdışı rota oluřturmada farklı türdeki senaryolar üzerinden Dijkstra algoritması ile karşılařtırıldığında çoęunlukla daha iyi sonuçlar elde edilebilmiřtir. Öte yandan, önerilen yaklaşımın çözüm süreleri deęerlendirildiğinde problem boyutu arttıkça çözüm sürelerinin de üstel olarak artacaęı görölmüřtür. Bu sebeple, çözüm sürelerini optimize etmek için paralel programlama vb. yöntemler uygulanabilir.



5. SONUÇLAR VE TARTIŞMA

Bu çalışmada trafik ile ilgili veri analitiği uygulamaları yapılmıştır. Trafikte yaşanan aksaklıklara ve yarattığı sorunlara değinilmiş ve bu sorunların çözümü için veri analitiklerine dayalı yaklaşımlar önerilmiştir.

Çalışmanın ikinci bölümünde trafik akışları için bir kestirimsel analitik yaklaşımı sunulmuş ve bir zaman serisi olarak ele alınan trafik akışı için özgün ve melez bir algoritma olan Yanlı rastgele anahtarlı genetik algoritma temelli bir Yapay Sinir Ağı modeli geliştirilmiştir. Gerçekleştirilen testlerde önerilen özgün algoritma ile başarılı sonuçlar elde edilebilmiştir.

Çalışmanın üçüncü bölümünde Eskişehir’de yaşanan trafik kazalarına yönelik tanımlayıcı analitik yaklaşımları uygulanmıştır. Gerçekleşen trafik kazalarının özelliklerini ortaya çıkarmak için bazı istatistiksel yöntemler, kümeleme ve öznitelik seçimi uygulanmıştır. Ayrıca, kazaları tahmin etmek için bir tek sınıf sınıflandırma yaklaşımı sunulmuştur. Literatürde daha önce tek sınıf sınıflandırma problemi olarak ele alınmayan trafik kaza tahmin problem için uygulanan testler sonucunda rekabetçi sonuçlar elde edilebilmiştir.

Çalışmanın son bölümünde trafik ağlarında optimal rota bulmak için bir kurallı analitik uygulaması olarak Markov Karar Süreci temelli bir dinamik programlama yaklaşımı sunulmuştur. Düşünülen trafik ağında yaşanabilecek aksamalardan dolayı kavşaklar arasındaki seyahat süreleri belirsizdir ve yaşanabilecek aksamalar yolların tamamen kapanmasına sebep olabilir. Trafikte yer alan trafik ışıkları da seyahat süresini etkiler. Tüm bu gecikme yaratabilecek durumlar göz önüne alınarak trafikte en kısa sürecek optimal rotayı bulabilmek için sezgisel bir yaklaşım önerilmiştir. Çevrimdışı olarak tasarlanan bu yaklaşımda çalışmanın önceki bölümlerinde uygulanan tahminler sonucu elde edilen bilgiler de sisteme dahil edilerek ağda yaşanabilecek aksamalar sonucu yol kapanma durumlarının önceden tahmin edildiği varsayılmıştır. Önerilen bu yaklaşım ile elde edilen rotalar, en kısa yol problemlerinde optimal çözümü veren Dijkstra algoritması ile elde edilen rotalarla farklı türdeki senaryolar üzerinden karşılaştırıldığında önerilen yöntemin bazı durumlar için baskın olduğu sonucuna varılmıştır.

Çalışmada trafik tek yön olarak varsayılarak modellenmiştir. Gelecek çalışma olarak yolların çift yön olarak ele alınıp rota oluşturulurken karşı yöne geçme durumları da modele dahil edilmelidir. Ayrıca Eskişehir örneğinde olduğu gibi tramvay gibi trafik akışını aksatabilecek daha fazla durum modele yansıtılabilir. Önemli günler, yolları etkileyen belirli olaylar da trafik akışını aksatmaktadır. İleride yapılacak çalışmalarda olay tabanlı bu durumların da rotalama modeline eklenerek gerçek trafik ağına daha fazla yaklaşmak planlanmaktadır.

Sonuç olarak bu çalışmada, trafik için bütünsel bir yaklaşım uygulanmıştır. Tahmin edilen trafik değişkenleri kullanılarak deterministik yöntemlere göre daha iyi sonuçlar elde edilebilmiştir. Önerilen bu çözüm yaklaşımı ambulans, polis, itfaiye gibi acil durum ekipleri ve ticari taksilerin rota önerisi vermek amacıyla kullanılabilir. Ayrıca, çevrimiçi bilgiler sağlayabilecek entegre bir sistem ile daha başarılı rotalar elde edilebilir. Uygulanan yaklaşım çevrimiçi bilgi toplama ve işleme sistemi de dahil edildiğinde bir akıllı trafik sistemi uygulaması olarak değerlendirilebilir.

KAYNAKÇA

- Abdel-Aty, M. A., ve Radwan, E. A. (2000). Modelling traffic accident occurrence and involvement. *Accident Analysis and Prevention*, 32(5), 633-642.
- Abdelwahab, H. T., ve Abdel-Aty, M. A. (2001). Development of artificial neural network models to predict driver injury severity in traffic accidents at signalized intersections. *Transportation Research Record*, 1746(1), 6-13.
- Acadgild. (2019, Aralık 1). *What are the different types of data analytics?* Acadgild: <https://acadgild.com/blog/different-types-of-data-analytics> (Erişim tarihi: 01.12.2019)
- Adhikari, R., ve Agrawal, R. (2013). *An Introductory Study on Time Series Modeling and Forecasting*. Lambert Academic Publishing.
- Ajah, I. A., ve Nweke, H. F. (2019). Big Data and Business Analytics: Trends, Platforms, Success Factors and Applications. *Big data and Congitive Computing*.
- Akgüngör, A. P., ve Doğan, E. (2010). Farklı yöntemler kullanılarak geliştirilen trafik kaza tahmin modelleri ve analizi. *Int. J. Eng. Research and Development*, 2(1) 16-22.
- Alkaabi, A. M., Disssanayake, D., ve Bird, R. (2011). Analyzing clearence time of urban traffic accidents in Abu Dhabi, United Arab Emirates, with hazard-based duration modeling method. *Journal of the Transportation Research Board*, 2229(1), 46-54.
- Alkheder, S., Taamneh, M., ve Taamneh, S. (2016). Severtiy prediction of traffic accident using an artificial neural network. *Journal of Forecasting*, 100-108.
- Alwaisi, S., ve Baykan, O. K. (2017). Training Of Artificial Neural Network Using Metaheuristic Algorithm. *International Journal of Intelligent Systems and Applications in Engineering*, Special Issue, 12-16.
- Bean, J. C. (1994). Genetic algorithms and random keys for sequencing and optimization. *ORSA Journal of Computing*, 6(2),154-160.
- Beshah, T., ve Hill, S. (2010). Mining road traffic accident data to impove safety: role of road-related factors on accident severity in Ethiopia. *AAAI Spring Symposium*.

- Brockwell, P. J., ve Davis, R. A. (2016). *Introduction to time series and forecasting*. Springer.
- Cai, L., Chen, Q., Cai, W., Xu, X., Zhou, T., ve Qin, J. (2019). SVRGSA: a hybrid learning based model for short-term traffic flow forecasting. *IET Intelligent Transport Systems*, 13(9), 1348-1355.
- Chan, K. Y., Dillion, T., Chang, E., ve Singh, J. (2013). Prediction of short-term traffic variables using intelligent swarm-based neural networks. *IEEE Transactions on Control Systems*, 21(1), 263-274.
- Chan, K. Y., Singh, J., Dillion, T. S., ve Chang, E. (2011). Traffic flow forecasting neural networks based on exponential smoothing method. *6th IEEE Conference on Industrial Electronics and Its Applications (ICIEA)*.
- Chandrashekar, G., ve Sahin, F. (2014). A survey on feature selection methods. *Computers and Electrical Engineering*, 40(1), 16-28.
- Chang, L. Y., ve Wang, H. W. (2006). Analysis of traffic injury severity: An application of non-parametric classification tree techniques. *Accident Analysis and Prevention*, 38(5), 1019-1027.
- Chen, J. F., Do, Q. H., ve Hsieh, H. N. (2015). Training Artificial Neural Networks by a Hybrid PSO-CS Algorithm, *Algorithms*, 8(2), 292-308.
- Chen, T., Zhang, C., ve Xu, L. (2016). Factor analysis of fatal road traffic crashes with massive casualties in China. *Advances in Mechanical Engineering*, 8(4), 1-11.
- Cheng, X., Ren, L., Cui, J., ve Zhang, Z. (2017). Traffic Flow Prediction with Improved SOPIO-SVR Algorithm. *Challenges and Opportunity with Big Data. Monterey Workshop 2016. Lecture Notes in Computer Science*. Springer.
- Chiroma, H., Noor, A. S., Abdulkareem, S., Abubakar, A. I., Hermawan, A., Qin, H., . . . Herawan, T. (2017). Neural Networks Optimization through Genetic Algorithm Searches: A Review. *Appl. Math. Inf. Sci*, 11(6), 1543-1564.
- Chung, Y. (2010). Development of an accident duration prediction model on the Korean Freeway Systems. *Accident Analysis and Prevention*, 42(1) 282-289.

- Conteras, J., Espinola, R., Nogales, F. J., ve Conejo, A. J. (2003). ARIMA models to predict next-day electricity prices. *IEEE Transactions on Power Systems*, 18(3), 1014-1020.
- Dener, M., Akcayol, M. A., Toklu, S., ve Bay, Ö. F. (2011). Zamana Bağlı Dinamik En Kısa Yol Problemi için Genetik Algoritma Tabanlı Yeni Bir Algoritma. *Gazi Üniv. Müh. Mim. Fak. Der.*, 26(4), 915-928.
- Dijkstra, E. W. (1959). A note on two problems in connexion with Graphs. *Numerische Mathematik*, 1, 269-271.
- Donate, J. P., Li, X., Sanches, G. G., ve Miguel, A. S. (2013). Time series forecasting by evolving artificial neural networks with genetic algorithms, differential evolution and estimation of distribution algorithm. *Neural Computing and Applications*, 22, 11-20.
- Dreisseitl, S. ve Ohno-Machado, L. (2002). Logistic regression and artificial neural network classification models: a methodology review. *Journal of Biomedical Informatics*, 35(5-6), 352-359.
- Emniyet Genel Müdürlüğü. (2018). Trafik İstatistik Bülteni. <http://trafik.gov.tr/istatistikler37> (Erişim tarihi: 01.01.2019)
- Emo, A. K., Matthews, G., ve Funke, G. J. (2016). The slow and the furious: Anger, stress and risky passing in simulated traffic congestion. *Transportation Research Part F: Traffic Psychology and Behaviour*, 42(1) 1-14.
- Eskişehir Büyükşehir Belediyesi. (2016). Eskişehir İstatistikleri. http://www.eskisehir.bel.tr/dosyalar/istatistiklerle_eskisehir/2016.pdf (Erişim tarihi: 01.12.2019)
- Givan, B., ve Parr, R. (2001). *An Introduction to Markov Decision Processes*. Purdue University ve Duke University.
- Gonçalves, J. F., ve Resende, M. G. (2011). Biased random-key genetic algorithms for combinatorial optimization. *Journal of Heuristics*, 17, 487-525.
- Habtemichael, F. G., ve Cetin, M. (2016). Short-term traffic flow rate forecasting based on identifying similar traffic patterns. *Transportation Research Part C*, 66, 61-78.

- Hart, P. E., Nilsson, N. J., ve Raphael, B. (1968). A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions of Systems Science and Cybernetics*, 4(2), 100-107.
- Highways England. *Traffic Flow Data*.
<http://tris.highwaysengland.co.uk/detail/trafficflowdata> (Eriřim Tarihi: 01.01.2018)
- Holt, C. C. (2004). Forecasting seasonals and trends by exponentially weighted moving averages. *International Journal of Forecasting*, 20(1), 5-10.
- Hu, L., Zhong, Y., Hao, W., Moghimi, B., Huang, J., Zhang, X., ve Du, R. (2018). Optimal Route Algorithm Considering Traffic Light and Energy Consumption. *IEEE Access*, 6, 59695-59704.
- Hu, W., Yan, L., Liu, K., ve Wang, H. (2016). A Short-term Traffic Flow Forecasting Method Based on the Hybrid PSO-SVR. *Neural Processing Letters*, 43, 155-172.
- Huang, Z. (1998). Extensions to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values. *Data Mining and Knowledge Discovering*, 2, 283-304.
- Hyndman, R. *Time Series Data Library*. (Eriřim Tarihi: 01.01.2018)
- Hyndman, R. J., ve Athanasopoulos, G. (2018). *Forecasting: Principles and Practice*. OTexts.
- Irani, R., ve Nasimi, R. (2011). Evolving neural network using real coded genetic algorithm for permeability estimation of the reservoir. *Expert Systems with Applications*, 38(8), 9862-9866.
- Isa, N., Mohamed, A., ve Yusoff, M. (2015). Implementation of Dynamic Traffic Routing for Traffic Congestion: A Review. *International Conference on Soft Computing in Data Science*.
- İsbak. (2018). *A Guide to Intelligent Transportation and Systems ve Best Practices*.
- Islam, B., Baharuddin, Z., Raza, M. Q., ve Nallagownden, P. (2014). Optimization of neural network architecture using genetic algorithm for load forecasting. *5th International Conference on Intelligent and Advanced Systems (ICIAS)*.

- Jain, A. K. (2010). Data clustering: 50 years beyond K-means. *Pattern Recognition Letters*, 31(8), 651-666.
- Karaboga, D., Akay, B., ve Ozturk, C. (2007). Artificial Bee Colony (ABC) Optimization Algorithm for Training Feed-Forward Neural Networks. *Modelling Decisions for Artificial Intelligence* Kitakyushu: Springer , 318-329.
- Keskin, M. V., ve Yıldız, D. (2018). Büyük veri araçları ve R kullanarak Amerikan havayolu firmalarının sorunlarının keşfedilmesi. *Uygulamalı Sosyal Bilimler Dergisi*, 2(2), 1-19.
- Kumar, V. S., ve Vanajakshi, L. (2015). Short-term traffic flow prediction using seasonal ARIMA model with limited input data. *Eur. Transp. Res. Rev*, 7(21).
- Kurt I., Ture M. ve Kurum. T. (2018). Comparing performances of logistic regression, classification and regression tree, and neural networks for predicting coronary artery disease. *Expert System with Applications*, 34(1), 366-374.
- Lin, L., Wang, Q., ve Sadek, A. (2015). A novel variable selection method based on frequent pattern tree for real-time traffic accident risk prediction. *Transportation Research Part C: Emerging Technologies*, 55, 444-459.
- Lord, D., ve Mannering, F. (2010). The statistical analysis of crash-frequency data: A review and assessment of methodological alternatives. *Transportation Research Part A: Policy and Practice*, 44(5), 291-305.
- Lv, Y., Tang, S., Zhao, H., ve Li, S. (2009). Real-time highway accident prediction based on support vector machines. *2009 Chinese Control and Decision Conference*.
- Madkour, A., Aref, W. G., Rehman, F. U., Rahman, M. A., ve Basalamah, S. (2017). *A Survey of Shortest-Path Algorithms*. Arxiv: <https://arxiv.org/pdf/1705.02044.pdf> (Erişim tarihi: 01.01.2020)
- Makridakis, S., Spiliotis, E., ve Assimakopoulos, V. (2018). Statistical and Machine Learning forecasting methods: concerns and ways forward. *PLoS ONE*, 13(3).
- Moayedi, H., Mehrabi, M., Mosallanezhad, M., Rashid, A. S., ve Pradhan, B. (2019). Modification of landslide susceptibility mapping using optimized PSO-ANN technique. *Engineering with Computers*, 35, 967-984.

- Molla, M. M. (2016). Identification of road traffic fatal crashes leading factors using principal component analysis. *International Journal of Research in Engineering and Technology*, 5(1), 79-84.
- Moreno, J. M., Pol, P. A., Abad, A. S., ve Blasco, B. C. (2013). Using the R-MAPE index as a resistant measure of forecast accuracy. *Psicothema*, 25(4), 500-506.
- Moretti, F., Pizzuti, S., Panzieri, S., ve Annunziato, M. (2015). Urban traffic flow forecasting through statistical and neural network bagging ensemble hybrid modeling. *Neurocomputing*, 167, 3-7.
- Murat, Y. Ş., ve Şekerler, A. (2009). Trafik kaza verilerinin kümeleme analizi yöntemi ile modellenmesi. *İMO Teknik Dergi*, 4759-4777.
- Narisimhan, G., Ephrem, B. G., Cheriyan, S., ve Balasupramanian, N. (2017). Predictive analytics of road accidents in Oman using machine learning approach. *International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, 1058-1065.
- Nicholson, A., ve Wong, Y. D. (1993). Are accidents poisson distributed? A statistical test. *Accident Analysis and Prevention*, 25(1) 91-97.
- Nur, A. S., Radzi, N. H., ve Ibrahim, A. O. (2014). Artificial Neural Network Weight Optimization: A Review. *TELKOMNIKA Indonesian Journal of Electrical Engineering* , 6987-6902.
- Ojha, V. K., Abraham, A., ve Snasel, V. (2017). Metaheuristic design of feedforward neural networks: a review of two decades of research. *Engineering Applications of Artificial Intelligence*, 60, 97-116.
- Panigrahi, S., Karali, Y., ve Behera, H. S. (2013). Time series forecasting using evolutionary neural network. *International Journal of Computer Applications*, 75(10), 13-17.
- Permanasari, A. E., Hidayah, I., ve Bustoni, I. A. (2013). SARIMA (Seasonal ARIMA) Implementation on Time Series Forecast The Number of Malaria Incidence. *International Conference on Information Technology and Electrical Engineering (ICTEE)*.

- Qi, L., ve Schneider, M. (2015). Traffalgorithm A Traffic-aware Shortest Path Algorithm in Real Road Networks with Traffic Influence Factors. *1st International Conference on Geographical Information Systems Theory Applications and Management (GISTAM)*, (s. 1-8).
- Rangel, H. R., Puig, V., Farias, R. L., ve Flores, J. J. (2017). Short-term demand forecast using a bank of neural network models trained using genetic algorithms for the optimal management of drinking water networks. *Journal of Hydroinformatics*, 19(1), 1-16.
- Reed, T., ve Kidd, J. (2019). *Global Traffic Scorecard*. Inrix. <http://inrix.com/press-releases/scorecard-2018-us/> (Eriřim tarihi: 01.01.2020)
- Reddy, H. (2013). Pathfinding—Dijkstra’s and A* Algorithm’s. , *International Journal in IT and Engineering*, 1-15.
- Rojas, R. (1996). *Neural Networks A Systematic Introduction*. Springer.
- Schniederjans, M. J., Schniederians, D. G., ve Starkey, C. M. (2014). *Bussiness Analytics Principles, Concepts, and Applications with SAS What, Why, and How*. Pearson Education.
- Schölkopf, B., Platt, J. C., Shawe-Taylor, J., ve Smola, A. J. (2001). Estimating the support of a high-dimensional distribution. *Neural Computation*, 13, 1443-1471.
- Sever, D., Zhao, L., Dellaert, N., Demir, E., Woensel , T. V., ve Kok, T. D. (2018). The dynamic shortest path problem with time-dependent stochastic disruptions. *Transportation Research Part C: Emerging Technologies*, 92, 42-57.
- Sharma, B., Katiyar, V. K., ve Kumar, K. (2015). Traffic Accident Prediction Model Using Support Vector Machines with Gaussian Kernel. *Proceedings of Fifth International Conference on Soft Computing for Problem Solving*.
- Shashank, B., Ballare, S., Rohit, ve Chandel, M. K. (2017). Impact of congestion on greenhouse gas emissions for road transport in Mumbai metropolitan region. *Transportation Research Procedia*, 25, 3538-3551.

- Shen, G., Kong, X., ve Chen, X. (2011). A short-term traffic flow intelligent hybrid forecasting model and its application. *Control Engineering and Applied Informatics*, 13(3), 65-73.
- Sokolova, M., Japkowicz, N., ve Szpahowicz, S. (2006). Beyond Accuracy, F-score and ROC: a Family of Discriminant Measures for Performance Evaluation. *Advances in Artificial Intelligence, 19th Australian Joint Conference on Artificial Intelligence*, 1015-1021.
- Song, H., Chen-Ching, L., Lawarrée, J., ve Dahlgren, R. W. (2000). Optimal Electricity Supply Bidding by Markov Decision Process. *IEEE Transactions on Power Systems*, 15(2), 618-624.
- Sun, Y., Yu, X., Bie, R., ve Song, H. (2017). Discovering time-dependent shortest path on traffic graph for drivers. *Journal of Network and Computer Applications*, 83(1) 204-212.
- Sunita, ve Garg, D. (2018). Dynamizing Dijkstra: A solution to dynamic shortest path problem through retroactive priority queue. *Journal of King Saud University - Computer and Information Sciences*.
- Taamneh, M., Alkheder, S., ve Taamneh, S. (2017). Data-mining techniques for traffic accident modelling and prediction in the United Arab Emirates. *Journal of Transportation Safety and Security*, 9(2), 144-166.
- Taccari, L. (2016). Integer programming formulations for the elementary shortest path problem. *European Journal of Operations Research*, 252(1) 122-130.
- Türkiye İstatistik Kurumu. *Ulaştırma istatistikleri*.
http://www.tuik.gov.tr/PreTablo.do?alt_id=1051, (Erişim tarihi: 01.01.2020)
- Vlahogianni, E. I., Karlaftis, M. G., ve Golias, J. C. (2005). Optimized and meta-optimized neural networks for short-term traffic flow prediction: A genetic approach. *Transportation Research Part C*, 13(3) 211-234.
- Wagstaf, K., Cardie, C., Rogers, S., ve Schroedl, S. (2001). Constrained K-means Clustering with Background Knowledge. *Proceedings of the Eighteenth International Conference on Machine Learning*.

- Wang, C., ve Ye, Z. (2015). Traffic flow forecasting based on a hybrid model. *Journal of Intelligent Transportation Systems*, 20, 428-437.
- Watanebe, K., ve Nakamura, H. (2016). A factor analysis of traffic accident frequency by types on major road sections in Japan. *Asian Transport Studies*, 4(2), 392-407.
- White, C. C. (2011). Markov decision processes. *Encyclopedia of Operations Research and Management Science*. Boston: Springer.
- Winters, R. (1960). Forecasting Sales by Exponentially Weighted Moving Averages. *Management Science*, 6(3) 324-342.
- Wu, P.-J., ve Yang, C.-K. (2017). The green fleet optimization model for a low-carbon economy: A prescriptive analytics. *International Conference on Applied System Innovation (ICASI)*, 107-110. Sapporo, Japan.
- Xiao, Y., Wang, H., ve Xu, W. (2015). Parameter Selection of Gaussian Kernel for One-Class SVM. *IEEE Transactions On Cybernetics*, 45(5), 927-939.
- Y., X., ve He, M. (2018). Improved artificial neural network based on intelligenet optimization algorithm. *Neural Network World*, 345-360.
- Yıldırım, U., ve Özcan, U. (2015). Ankara şehir içi trafik kazalarının analizi: bir yapay sinir ağı modeli. *Gazi Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 17(1), 35-63.
- Yu, B., Wang, Y. T., Yao, J. B., ve Wang, J. Y. (2016). A comparision of the performance of ANN and SVM for the prediction of traffic accident duration. *Neural Network World*, 26(3), 271-287.
- Yu, R., ve Li, Y. (2017). Deep Learning: A Generic Approach for Extreme Condition Traffic Forecasting. *Proceedings of the 2017 SIAM International Conference on Data Mining*, 777-785.
- Yuan, Z., Zhou, X., Yang, T., ve Tamer, J. (2017). Predicting Traffic Accidents Through Heterogeneous Urban Data: A Case Study. *Proceedings of 6th International Workshop on Urban Computing*,. Halifax.
- Zhang, G. P. (2003). Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing*, 50, 159-175.

- Zhang, G., Patuwo, B. E., ve Hu, M. Y. (1998). Forecasting with artificial neural networks:: The state of the art. *International Journal of Forecasting*, 14(1), 35-62.
- Zhang, Z., He, Q., Gao, J., ve Ni, M. (2018). A deep learning approach for detecting traffic accidents from social media data. *Transportation Research Part C: Emerging Technologies*, 86, 580-596.
- Zhou, J., Chan, K. L., Chong, V. F., ve Krishnan, S. M. (2005). Extraction of Brain Tumor from MR Images Using One-Class Support Vector Machine. *IEEE Engineering in Medicine and Biology 27th Annual Conference*. Shanghai, China.



ÖZGEÇMİŞ

Ad Soyad: Zeynep İdil ERZURUM ÇİÇEK

Doğum Yeri ve Yılı: İstanbul / 1988

E-posta: zierzurum@eskisehir.edu.tr

Eğitim:

- 2006 – 2011, Lisans, Anadolu Üniversitesi, Mühendislik Fakültesi, Endüstri Mühendisliği Bölümü
- 2011 – 2015, Yüksek lisans, Anadolu Üniversitesi, Fen Bilimleri Enstitüsü, Endüstri Mühendisliği Anabilim Dalı
- 2015 – 2020, Doktora, Eskişehir Teknik Üniversitesi, Lisansüstü Eğitim Enstitüsü, Endüstri Mühendisliği Anabilim Dalı

Mesleki Deneyim:

- 2011 – 2012, Üretim Planlama ve Kontrol Uzman Yardımcısı, Eczacıbaşı Yapı Gereçleri Artema, Bozüyük, Bilecik
- 2012 – 2018, Araştırma Görevlisi, Anadolu Üniversitesi, Endüstri Mühendisliği Bölümü.
- 2012 – 2015, Araştırma Görevlisi, Anadolu Üniversitesi, Bilgisayar Araştırma ve Uygulama Merkezi
- 2018 – ... , Araştırma Görevlisi, Eskişehir Teknik Üniversitesi, Endüstri Mühendisliği Bölümü.

Uluslararası hakemli dergilerde yayımlanan makaleler:

1. Erzurum Cicek Z.I., ve Kamisli Ozturk, Z. (2018). Short term traffic flow forecasting using artificial neural networks. *Sigma Journal of Engineering and Natural Sciences*, 4(9), 405-414.
2. Kamisli Ozturk, Z., Erzurum Cicek, Z.I., ve Ergul, Zeliha (2017). Sentiment Analysis: an Application to Anadolu University. *Acta Physica Polonica A*, 132(3), 753-755., Doi: 10.12693/APhysPolA.132.753

Uluslararası bilimsel toplantılarda sunulan ve bildiri kitaplarında basılan bildiriler :

1. Erzurum Cicek Z.I., ve Kamisli Ozturk, Z. (2019). Combination Methods for Traffic Flow Forecasting. 10th International Symposium on Intelligent Manufacturing and Service Systems (Tam Metin Bildiri/Sözlü Sunum)
2. Kamisli Ozturk, Z. ve Erzurum Cicek, Z.I. (2019). Identification Key Factors of Traffic Accidents Using Data Mining. 10th International Symposium on Intelligent Manufacturing and Service Systems (Tam Metin Bildiri/Sözlü Sunum)
3. Erzurum Cicek Z.I., ve Kamisli Ozturk Z. (2017). A mathematical model for a multi objective single machine scheduling problem with sequence dependent setup times and due dates. *International Conference on Engineering Technologies* (Özet Bildiri/Sözlü Sunum)
4. Erzurum Cicek Z.I., ve Kamisli Ozturk, Z. (2017). Short Term Traffic Flow Forecasting Using Artificial Neural Networks. *TRANSIST 2017* (Tam Metin Bildiri/Sözlü Sunum)
5. Erzurum Cicek Z.I., ve Kamisli Ozturk, Z. (2017). A comparison of scalarization techniques on multi-objective machine schedulingproblem with sequence dependent setup times. *International Conference On Mathematics And Engineering* (Özet Bildiri/Sözlü Sunum)
6. Erzurum Cicek, Z.I., Kamisli Ozturk, Z. ve Ergul, Z. (2016). Sentiment analysis: An application to Anadolu University. *ICCESEN* (Özet Bildiri/Sözlü Sunum)
7. Kamisli Ozturk, Z., Ergul Z. ve Erzurum Cicek, Z.I. (2016). Social Media Mining for Open and Distance Education Systems. *EURO* (Özet Bildiri/Sözlü Sunum)

Yazılan ulusal/uluslararası kitaplar veya kitaplardaki bölümler:

1. Erzurum Cicek Z.I., ve Kamisli Ozturk, Z. (2017). Societal Complexity, Data Mining and Gaming State-of-the-Art 2017, Bölüm adı:(The Future of IoT and Big Data Analytics in Traffic Problems), Amsterdam Europe: Greenhill Waterfront, Editör:Dorien DeTombe, Gerhard-Wilhelm Weber and Semih Kuter, Basım sayısı:1, ISBN:978-90-77171-54-7, İngilizce(Bilimsel Kitap)

Ulusal bilimsel toplantılarda sunulan ve bildiri kitaplarında basılan bildiriler:

1. Köksal B., Ergul Z., Erzurum Cicek, Z.I. ve Kamisli Ozturk Z. (2019). Yurtdışı Eğitim Kararı İçin Çok Aşamalı Çözüm Yaklaşımı. 9. Yöneylem Araştırması / Endüstri Mühendisliği Ulusal Kongresi (Özet Bildiri/Sözlü Sunum)
2. Ergul Z, Kamisli Ozturk, Z. ve Erzurum Cicek, Z.I. (2018). Comparison of Machine Learning Algorithms on a Sentiment Analysis Application. 38. Yöneylem Araştırması / Endüstri Mühendisliği Ulusal Kongresi (Özet Bildiri/Sözlü Sunum)
3. Erzurum Cicek Z.I., ve Kamisli Ozturk, Z.(2018). Trafikte Tahmin ve Veri Analitiği. 38. Yöneylem Araştırması Endüstri Mühendisliği Ulusal Kongresi(YAEM 2018) (Özet Bildiri/Sözlü Sunum)

