



**MAKİNE ÖĞRENME VE PEKİŞTİRMELİ ÖĞRENME ALGORİTMALARI
İLE DENİZALTI SAVUNMA HARBİNİN MODELLENMESİ**

HAKAN AKYOL

ARALIK 2023

ÇANKAYA ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

VERİ ANALİTİĞİ ANA BİLİM DALI

Yüksek Lisans

VERİ ANALİTİĞİ

**MAKİNE ÖĞRENME VE PEKİŞTİRMELİ ÖĞRENME ALGORİTMALARI
İLE DENİZALTI SAVUNMA HARBİNİN MODELLENMESİ**

HAKAN AKYOL

ARALIK 2023

ÖZET

MAKİNE ÖĞRENME VE PEKİŞTİRMELİ ÖĞRENME ALGORİTMALARI İLE DENİZALTI SAVUNMA HARBİNİN MODELLENMESİ

AKYOL, HAKAN

Veri Analitiği Yüksek Lisans

Danışman: Prof. Dr. Hayri SEVER

Aralık 2023 , 139 sayfa

Denizaltılar savaş sahnesine çıktıkları ilk andan itibaren savaşın gidişatını doğrudan etkileyebilen platformlar olmuştur. Bu platformlar dost unsur olarak kullanıldığında büyük bir kuvvet çarpanı olabilecekleri gibi tehdit unsur olduklarında ise büyük bir risk olacaktırlar. Bu nedenle harekât sahasında bu denli büyük etkiye sahip platformlara karşı etkin bir savunma yapısına sahip olunması oldukça önemlidir. Bu çalışmada tehdit denizaltılara karşı etkin bir savunma yapısına sahip olunabilmesi için gerekli platformlar (savaş gemisi, uçak ve helikopter) ile bu platformlardan atılacak silah sayıları hesaplanmaya çalışılmıştır. Bu kapsamda ilk olarak çalışmada kullanılacak veriler tanımlanmış ve anılan veriler ön işleme tabi tutulmuştur. Bu doğrultuda DSH'ı etkileyen parametrelerin neler olduğu belirlenmiş ve bu parametreler için uygun dağılımlara sahip verilerden 220 satırlık kaynak veri seti oluşturulmuştur. Daha sonra bu veri setindeki veri sayılarını artırmak için 8 farklı sentetik veri üretme tekniğinden faydalanarak (her bir teknik ile veri setindeki satır sayısı 10000 yapılmıştır.) yeni veri setleri oluşturulmuştur. Müteakiben hangi teknik ile veri artırımı yapılacağının belirlenmesi için oluşturulan veri setlerine bir dizi test uygulanarak en iyi sonucu veren sentetik veri üretme tekniği belirlenmiştir. Yapılan inceleme neticesinde en uygun veri artırım yönteminin MDO yöntemi olduğu belirlenmiştir.

Ardından MDO yöntemiyle oluşturulmuş 10000 adet veri ile 7 farklı makine öğrenme modeli kullanılarak her bir tehdit denizaltı imhası için gerekli silah sayıları belirlenmiştir. Çalışmanın sonraki bölümünde; ilk olarak tehdit denizaltı tespit olasılıkları hesaplanmıştır. Ardından tespit olasılıkları, atılması gereken silah sayıları ve platform/silah maliyetleri kullanılarak ve Pekiştirmeli Öğrenme algoritmasından faydalanılarak tüm tehdit denizaltıların imhası için çeşitli harekât tarzlarından alınan sonuçlar hesaplanmıştır. Bu hesaplamalara göre optimal harekât tarzı belirlenmiştir. Çalışmanın son kısmında ise elde edilecek kazanımlara değinilmiştir.

Anahtar Kelimeler: Denizaltı Savunma Harbi, Sentetik Veri Üretme, Makine Öğrenme, Pekiştirmeli Öğrenme, Veri Artırma



ABSTRACT

MODELING ANTI SUBMARINE WARFARE USING MACHINE LEARNING AND REINFORCEMENT LEARNING ALGORITHMS

AKYOL, HAKAN

M.Sc. in Data Analytics

Supervisor: Prof. Dr. Hayri SEVER

December 2023, 139 pages

Submarines have been platforms that can directly affect the course of war from the moment they first appeared on the battlefield. While these platforms can be a great force multiplier when used as a friendly element, they will be a great risk when they are used as a threat. For this reason, it is very important to have an effective defense structure against platforms that have such a great impact in the field of operations. In this study, the necessary platforms (warship, aircraft and helicopter) and the number of weapons to be fired from these platforms were tried to be calculated in order to have an effective defense structure against threat submarines. In this context, firstly, the data to be used in the study were defined and the said data were pre-processed. In this regard, the parameters affecting submarine defense warfare were determined and a 220-line source data set was created from data with appropriate distributions for these parameters. Then, new data sets were created by using 8 different synthetic data generation techniques (the number of rows in the data set was increased to 10000 with each technique) to increase the number of data in this data set. Subsequently, a series of tests were applied to the created data sets to determine which technique would be used to increase data, and the synthetic data generation technique that gave the best results was determined.

As a result of the examination, it was determined that the most suitable data augmentation method was the MDO method. Then, the number of weapons required to destroy each threat submarine was determined by using 10000 data created by the MDO method and 7 different machine learning models. In the next part of the study; First, threat submarine detection probabilities were calculated. Then, using detection probabilities, number of weapons to be fired and platform/weapon costs, and using the Reinforcement Learning algorithm, the results obtained from various operational modes were calculated to destroy all threat submarines. According to these calculations, the optimal mode of operation was determined. In the last part of the study, the gains to be achieved are mentioned.

Keywords: Anti Submarine Warfare, Data Augmentation, Machine Learning, Reinforcement Learning, Synthetic Data Generation

TEŞEKKÜR

Bu tez çalışmaları süresince her zaman yanımda olan her daim sabır ve destek gösteren değerli eşim Demet ISLAK AKYOL'a, çalışmalarım süresince teknik anlamda desteğini esirgemeyen kıymetli arkadaşım Yüksek Mühendis Ragıp ZİLCİ'ye ve tüm tez sürecinde çalışmamı her zaman daha da iyi seviyelere çıkarmamı sağlayan, bilgi birikimi ile gelişmemde büyük etkisi olan değerli tez danışmanım Prof.Dr. Hayri SEVER'e teşekkür ederim.

İÇİNDEKİLER

İNTİHAL OLMADIĞINA DAİR BEYAN SAYFASI	iii
ÖZET.....	iv
ABSTRACT	vi
TEŞEKKÜR	viii
TABLolar LİSTESİ.....	xii
ŞEKİLLER LİSTESİ.....	xiii
SİMGELER VE KISALTMALAR LİSTESİ	xvi
BÖLÜM I	1
GİRİŞ	1
1.1 DENİZALTILARIN SAVAŞ TARİHİNDEKİ YERİ.....	1
1.1.1 Tarihteki İlk Denizaltı	1
1.1.2 Denizaltıların Savaş Sahnesine Çıkışı	2
1.2 DSH'IN ÖNEMİ	4
1.3 LİTERATÜR TARAMASI	5
1.4 PROBLEMİN TANIMLANMASI	8
1.4.1 Problem 1	8
1.4.2 Problem 2.....	8
1.5 ÇALIŞMA YÖNTEMİ.....	9
BÖLÜM II	10
VERİ TANIMI VE ÖN İŞLEME.....	10
2.1 VERİLERİN TANIMLANMASI	10
2.1.1 Dost Kuvvet Verileri	10
2.1.1.1 Parametre Açıklamaları	10
2.1.2 Düşman Kuvvet Verileri	11
2.1.3 Çevresel Veriler.....	12
2.1.3.1 Parametre Açıklamaları	12
2.2 SENTETİK VERİLERİN ÜRETİLMESİ	18
2.2.1 Sentetik Veri Üretme Tekniklerinin İncelenmesi	18
2.2.1.1 SMOTE Yöntemi	18

2.2.1.2 Borderline-SMOTE Yöntemi.....	19
2.2.1.3 ADASYN Yöntemi	20
2.2.1.4 ROSE Yöntemi.....	20
2.2.1.5 Trim-SMOTE Yöntemi	21
2.2.1.6 MWMOTE Yöntemi	22
2.2.1.7 HAC Yöntemi	23
2.2.1.8 MDO Yöntemi.....	23
2.2.2 Sentetik Veri Üretme Tekniklerinin Karşılaştırılması.....	29
2.2.2.1 Histogram ve Yoğunluk Grafiklerinin İncelenmesi	29
2.2.2.2 Q-Q Çizgilerinin İncelenmesi	46
2.2.2.3 Kolmogorov-Smirnov Testinin Uygulanması	79
2.2.2.4 Accuracy Skorlarının İncelenmesi	83
2.2.2.5 F1 Skorlarının İncelenmesi	83
2.2.2.6 Keskinlik Değerlerinin İncelenmesi.....	84
2.2.2.7 Duyarlılık Değerlerinin İncelenmesi.....	84
2.2.2.8 ROC-AUC Değerlerinin İncelenmesi	85
BÖLÜM III.....	87
ATILACAK MÜHİMMAT SAYILARI VE TEHDİT DENİZALTI TESPİT	
OLASILIKLARININ HESAPLANMASI	87
3.1 ATILACAK TORPİDO SAYILARININ HESAPLANMASI.....	87
3.1.1 Makine Öğrenmesinin İncelenmesi	87
3.1.2 Çalışmada Kullanılan Makine Öğrenme Algoritmalarının İncelenmesi....	88
3.1.2.1 KNN Algoritması	88
3.1.2.2 Karar Ağaçları Algoritması	89
3.1.2.3 Rastgele Orman Algoritması.....	90
3.1.2.4 Gradient Boosting Algoritması	91
3.1.2.5 ANN	92
3.1.2.6 Stacking Yöntemiyle Oluşturulmuş Mix Algoritma	93
3.1.2.7 Voting Yöntemiyle Oluşturulmuş Mix Algoritma.....	94
3.1.3 Algoritma Performanslarının Karşılaştırılması ve En İyi Performans Gösteren Algoritmanın Belirlenmesi	95
3.1.4 Hedefin İmhası İçin Atılması Gerekli Mühimmat Sayılarının Hesaplanması	98

3.2 TEHDİT DENİZALTI TESPİT OLASILIKLARININ HESAPLANMASI ..	100
BÖLÜM IV	102
EN İYİ HAREKAT TARZI VE PLATFORM/MÜHİMMAT SAYILARININ	
HESAPLANMASI	102
4.1 PEKİŞTİRMELİ ÖĞRENME ALGORİTMASININ İNCELENMESİ.....	102
4.2 ÖDÜL/CEZA DEĞERLERİNİN HESAPLANMASI	103
4.3 HİPERPARAMETRE OPTİMİZASYONU.....	104
4.4 OPTİMAL HAREKÂT TARZININ BELİRLENMESİ.....	107
4.5 TOPLAM MÜHİMMAT SAYILARININ HESAPLANMASI.....	110
BÖLÜM V.....	111
SONUÇ VE DEĞERLENDİRME.....	111
5.1 MEVCUT YAKLAŞIMLARIN İNCELENMESİ.....	111
5.2 BU ÇALIŞMA İLE ELDE EDİLEN KAZANIMLAR	111
5.3 MÜTEAKİP DÖNEMDE YAPILABİLECEK ÇALIŞMALAR.....	112
KAYNAKÇA	113
EKLER.....	120
EK-1 HESAPLAMA METODOLOJİSİNİN GÖSTERİMİ	120
EK-2 Q-ÖĞRENME ALGORİTMASINA AİT SÖZDE KOD (PSEUDO-CODE)	
.....	121

TABLÖLAR LİSTESİ

Tablo 1: Literatürde Yer Alan Bazı Benzer Çalışmalar	5
Tablo 2: Çalışmada Kullanılan Dost Kuvvet Verileri.....	15
Tablo 3: Çalışmada Kullanılan Düşman Kuvvet Verileri	16
Tablo 4: Çalışmada Kullanılan Çevresel Veriler	16
Tablo 5: Dost Kuvvet Angajman Veri Tabanı Kaynak Veri Seti Örnek Gösterimi ...	17
Tablo 6: Sentetik Veri Üretme Tekniklerinin Olumlu ve Olumsuz Yönleri	25
Tablo 7: Kolmogorov Simirnov Testi Sonuçları.	80
Tablo 8: Doğruluk Skoru Değerleri.....	83
Tablo 9: F1 Skoru Değerleri.....	84
Tablo 10: Kesinlik Değerleri.....	84
Tablo 11: Duyarlılık Değerleri.....	85
Tablo 12: ROC-AUC Değerleri.....	85
Tablo 13: Makine Öğrenme Algoritmaları Tahmin Doğruluk Oranları.....	95
Tablo 14: Dost Kuvvet Tahmin Veri Seti Örnek Gösterimi.....	97
Tablo 15: Dost Kuvvet Tarafından Atılması Gereken Mühimmat Sayıları.....	98
Tablo 16: Düşman Kuvvet Tarafından Atılması Gereken Mühimmat Sayıları.....	99
Tablo 17: Tehdit Denizaltı Tespit Olasılıkları.....	101
Tablo 18: Hesaplama Kullarılan Platform Bilgileri.....	107
Tablo 19: Dost Kuvvet Optimal Harekât Tarzı.....	109
Tablo 20: Düşman Kuvvet Optimal Harekât Tarzı.....	109
Tablo 21: Toplam Mühimmat Stok Miktarları.....	110
Tablo 22: Tehdide Yönelik Olasılıksal Yaklaşım ile Hesaplanan Toplam Mühimmat Stok Miktarları.....	111

ŞEKİLLER LİSTESİ

Şekil 1: Drebbel'in Denizaltısı	1
Şekil 2: H.L. Hunley Denizaltısı	3
Şekil 3: Alman U-20 Denizaltısı.....	4
Şekil 4: SMOTE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-1	30
Şekil 5: SMOTE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-2	31
Şekil 6: Borderline SMOTE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-1.....	32
Şekil 7: Borderline SMOTE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-2.....	33
Şekil 8: ADASYN Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-1	34
Şekil 9: ADASYN Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-2.....	35
Şekil 10: ROSE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-1.	36
Şekil 11: ROSE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-2.	37
Şekil 12: Trim-SMOTE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-1.....	38
Şekil 13: Trim-SMOTE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-2.....	39
Şekil 14: MWMOTE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-1.....	40
Şekil 15: MWMOTE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-2.....	41
Şekil 16: HAC Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-1.	42

Şekil 17: HAC Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-2.	43
Şekil 18: MDO Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-1.	44
Şekil 19: MDO Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-2.	45
Şekil 20: SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-1.	47
Şekil 21: SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-2.	48
Şekil 22: SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-3.	49
Şekil 23: SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-4.	50
Şekil 24: Borderline SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-1.	51
Şekil 25: Borderline SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-2.	52
Şekil 26: Borderline SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-3.	53
Şekil 27: Borderline SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-4.	54
Şekil 28: ADASYN Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-1.	55
Şekil 29: ADASYN Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-2.	56
Şekil 30: ADASYN Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-3.	57
Şekil 31: ADASYN Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-4.	58
Şekil 32: ROSE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-1.	59
Şekil 33: ROSE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-2.	60
Şekil 34: ROSE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-3.	61
Şekil 35: ROSE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-4.	62
Şekil 36: Trim-SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-1.	63
Şekil 37: Trim-SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-2.	64
Şekil 38: Trim-SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-3.	65
Şekil 39: Trim-SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-4.	66
Şekil 40: MWMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-1.	67
Şekil 41: MWMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-2.	68
Şekil 42: MWMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-3.	69
Şekil 43: MWMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-4.	70
Şekil 44: HAC Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-1.	71
Şekil 45: HAC Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-2.	72
Şekil 46: HAC Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-3.	73
Şekil 47: HAC Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-4.	74

Şekil 48: MDO Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-1.	75
Şekil 49: MDO Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-2.	76
Şekil 50: MDO Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-3.	77
Şekil 51: MDO Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-4.	78
Şekil 52: Dost Kuvvet Yineleme Sayısı-Ödül Değeri Grafiği.	105
Şekil 53: Düşman Kuvvet Yineleme Sayısı-Ödül Değeri Grafiği.	105
Şekil 54: Dost Kuvvet Öğrenme Oranı-Ödül Değeri Grafiği.	106
Şekil 55: Düşman Kuvvet Öğrenme Oranı-Ödül Değeri Grafiği.	106
Şekil 56: Dost Kuvvet Keşif Oranı-Ödül Değeri Grafiği.	106
Şekil 57: Düşman Kuvvet Keşif Oranı-Ödül Değeri Grafiği.	107
Şekil 58: Hesaplama Dikkate Alınan Harekât Ortamının Görselleştirilmesi.....	108

SİMGELER VE KISALTMALAR LİSTESİ

Simgeler

C^0	: Santigrat Derece
%	: Yüzde
m^3	: Metre Küp
p_i^j	: i dost platformun j tehdit platformu tespit etme olasılığı.
w_i	: i dost platformun sonar menzili.
v_i	: i dost platformun sürati.
v_j	: j tehdit platformun sürati.
t_j	: j tehdit platformun sualtında kalma süresi.
r_k^d	: k durumunda dost kuvvetin kazandığı ödül değeri.
c_k^{td}	: k durumunda imha edilen düşman kuvvet D/A maliyeti.
c_k^d	: k durumunda imha edilen dost kuvvet platform maliyeti.
c_k^{dt}	: k durumunda imha edilecek D/A'nın imhası için dost kuvvet tarafından atılan silahın maliyeti.
a_k^{dt}	: k durumunda imha edilecek D/A'nın imhası için dost kuvvet tarafından atılan silah sayısı.
r_k^t	: k durumunda düşman kuvvetin kazandığı ödül değeri.
c_k^{tt}	: k durumunda imha edilecek dost kuvvet platformunun imhası için düşman kuvvet tarafından atılan silahın maliyeti.
a_k^{tt}	: k durumunda imha edilecek dost kuvvet platformunun imhası için düşman kuvvet tarafından atılan silah sayısı.

Kısaltmalar

DSH	: Denizaltı Savunma Harbi
Hz	: Hertz
kg	: Kilogram
dBA	:StandartDesibel

SMOTE	: Sentetik Azınlık Aşırı Örneklememe Tekniğı (Synthetic Minority Over Sampling Technique SMOTE)
ADASYN	: Uyarlamalı Sentetik Örneklememe (Adaptive Synthetic Sampling, ADASYN)
ROSE	: Rastgele Aşırı Örneklememe Örnekleri (Random Over Sampling Examples,)
MWMOTE	: Çoğunluk Ağırlıklı Azınlık Aşırı Örneklememe Tekniğı (Majority Weighted Minority Oversampling Technique,)
HAC	: Hiyerarşik Aglomeratif Kümeleme (Hierarchical Agglomerative Clustering,)
MDO	: Karışım Yoğunluğu Aşırı Örneklememe (Mixture Density Oversampling,)
GMM	: Gaussian Karışım Modeli (Gaussian Mixture Model)
EM	: Expectation-Maximization
CDF	: Kumulatif Dağılım Fonksiyonu (Cumulative Distribution Function)
ROC	: Alıcı İşletim Karakteristiğı (Receiver Operating Characteristic)
AUC	: Eğrinin Altındaki Alan (Area Under the Curve)
KNN	: K En Yakın Komşu (K-Nearest Neighbor)
ANN	: Yapay Sinir Ağları (Artificial Neural Networks)
MSE	: Hata Kareleri Ortalaması (Mean Squared Error)
D/A	: Denizaltı
MDP	: Markov Karar Süreci (Markov Decision Process)
SUH	: Suüstü Harbi (SUH)

BÖLÜM I

GİRİŞ

1.1 DENİZALTI LARIN SAVAŞ TARİHİNDEKİ YERİ

1.1.1 Tarihteki İlk Denizaltı

Cornelius Drebbel, 17. yüzyılın başında yaşamış olan Hollandalı bir mucittir. Drebbel, birçok buluşa imza atmış olsa da en ünlü icadı denizaltısıdır. Drebbel'in denizaltısı, zamanının ötesinde bir tasarıma sahiptir ve denizaltı teknolojisinin gelişmesine büyük katkıda bulunmuştur. Drebbel'in denizaltısı, 1620'lerde İngiltere'de inşa edilmiştir. Bu dönemde denizaltı teknolojisi henüz gelişmemiştir ve Drebbel'in tasarımı olağanüstü bir ilerleme olarak kabul edilmiştir. Denizaltı, ahşap bir gövdeye ve altı kürekçi tarafından hareket ettirilen küreklerle donatılmıştır. Ayrıca, içerideki hava tüpleri sayesinde dışarıdan havalandırma kabiliyetine sahiptir [1, 2, 3].



Şekil 1: Drebbel'in Denizaltısı

Drebbel'in denizaltısı, su altında ilerlemek için düşük basınç sistemi kullanan ilk denizaltılardan birisidir. Deniz suyunun dışarı sızmasını önlemek için conta ve kauçuk malzemeler kullanılmıştır. Denizaltı, yaklaşık 12 metre uzunluğunda ve 4 metre genişliğindedir. İçerisinde ise 12 kişilik bir mürettebatı taşıyabilecek kadar alana sahiptir [1, 2, 3].

Drebbel, denizaltısını 1620'lerde Thames Nehri'nde birkaç deneme seferine çıkarmıştır. Denizaltı başarılı bir şekilde su altında seyahat edebilmiş ve yüzeye çıkabilmiştir. Denizaltıdaki tayfun sistemi, su altında hava basıncını dengelemek için kullanılan önemli bir mekanizmadır. Bu sistemin çalışması sayesinde denizaltıda yaşayan insanlar oksijen ihtiyaçlarını karşılayabiliyordu [1, 2, 3].

Drebbel'in denizaltısı, su üstünde olduğu kadar su altında da seyahat edebilecek şekilde tasarlanmıştır. Bu özelliği, denizaltıyı çeşitli amaçlar için kullanılabilir kılmış, örneğin keşif, savaş veya su altı dünyasını gözlemleme gibi. Denizaltı, döneminde büyük bir ilgi uyandırmış ve çeşitli kraliyet gösterilerinde sergilenmiştir. Drebbel'in icadı, denizaltı teknolojisinin gelecekteki gelişimini etkileyen bir dönüm noktası olmuştur [1, 2, 3]. Drebbel'in denizaltısı, o dönemdeki denizaltı teknolojisine önemli bir temel sağlamıştır. İlerleyen yıllarda denizaltılar daha da geliştirilerek daha uzun süreler su altında kalabilen ve daha verimli hareket edebilen araçlara dönüşmüştür. Drebbel'in denizaltısı, denizaltı teknolojisinin temellerini atan önemli bir icattır ve gelecek nesiller için denizaltı gelişimine ilham vermiştir. Denizaltılar, günümüzde de askeri ve sivil amaçlar için yaygın olarak kullanılan bir araç haline gelmiştir.

1.1.2 Denizaltıların Savaş Sahnesine Çıkışı

H.L. Hunley, Amerikan İç Savaşı sırasında kullanılan ilk başarılı saldırı denizaltısıdır. İlk olarak 1863 yılında Güney Konfederasyonu tarafından inşa edilen bir denizaltıdır. Hunley, ismini dönemin Güneyli subaylarından Horace L. Hunley'den almıştır. 17 Şubat 1864 tarihinde, H.L. Hunley, Birleşik Devletler Deniz Kuvvetleri'ne ait USS Housatonic adlı savaş gemisini batırmak üzere Charleston, Güney Karolina'dan denize açıldı. Hunley, torpidosunu (denizaltı torpidosu) Housatonic'e karşı başarıyla kullanarak gemiyi batırmıştır. Bu, bir denizaltının bir savaş gemisini batırdığı ilk belgelenmiş olaydır [4, 5].



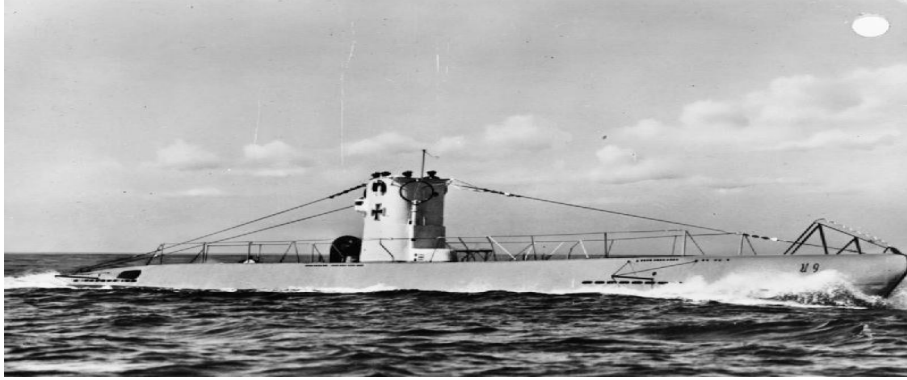
Şekil 2: H.L. Hunley Denizaltısı

USS Housatonic'i batırdıktan sonra, Hunley tekrar limana dönmek üzereyken, bilinmeyen bir sebepten dolayı suya batmıştır. Denizaltı ve mürettebatının tamamı kaybolmuştur. Hunley'nin neden battığı tam olarak belirlenememiştir, ancak bazı teoriler arasında torpidonun patlaması, su alması veya karbondioksit birikimi bulunmaktadır. H.L. Hunley'nin kaybolmasından sonra, denizaltının ve mürettebatının akıbetini belirlemek için arama ve kurtarma çalışmaları başlatıldı. Ancak Hunley'nin enkazı, 1995 yılına kadar bulunamamıştır. 1995 yılında keşfedilen Hunley, 2000 yılında çıkarılmış ve 2002 yılında kapsamlı bir restorasyon sürecine tabi tutulmuştur [4, 5].

Bilim adamları ve arkeologlar, denizaltının yapısını ve içerisindeki kalıntıları dikkatlice incelerken, Hunley'nin tarihi ve tasarımı hakkında daha fazla bilgi edinildi. Restorasyon çalışmaları tamamlandıktan sonra, H.L. Hunley 2004 yılında Charleston, Güney Karolina'da bir müze olarak halka açıldı ve ziyaretçilere sergilenmeye başlandı. H.L. Hunley'nin başarılı bir saldırı gerçekleştirmesi, denizaltı teknolojisinin gelecekteki savaşlarda ne kadar etkili olabileceğini göstermesi açısından önemlidir. Hunley'nin başarılı bir şekilde bir savaş gemisini batırması, denizaltıların geliştirilmesi ve kullanımının hızlanmasına yol açmıştır. Aynı zamanda, sivil gemilere karşı kullanılan bu tür saldırılar, sivil kayıplar ve uluslararası tepkilere yol açmış ve denizaltı savaşlarının etik ve hukuki boyutlarını gündeme getirmiştir.

Denizaltıların saldırı maksatlı kullanımına ilişkin başka bir örnekte Alman U botlarıdır. 7 Mayıs 1915 tarihinde, Alman İmparatorluk Donanması'na ait U-20 denizaltısı, İrlanda açıklarında seyreden ve yolcu taşıyan İngiliz gemisi RMS Lusitania'ya saldırmıştır. RMS Lusitania, 1907 yılında hizmete giren ve o dönemdeki en büyük yolcu gemisi olarak bilinen bir transatlantiktir. Almanlar, Lusitania'nın savaş malzemeleri taşıdığı iddiasıyla saldırıyı gerçekleştirmiştir. Ancak Lusitania,

sivil yolcuları taşıyan bir yolcu gemisiydi ve Almanlar bu gemiyi batırarak 1.198 kişinin hayatını kaybetmesine sebep olmuştur [6, 7, 8].



Şekil 3: Alman U-20 Denizaltısı

Lusitania'nın batırılması, uluslararası toplumda büyük bir tepkiye neden olmuştur. Özellikle Amerika Birleşik Devletleri, Almanya'ya karşı sert bir tepki göstermiş ve karşı saldırıya geçmiştir. Bu olay Amerika'nın savaşa girmesinde etkili bir faktör olmuştur. Ancak Amerika, Lusitania'nın batırılmasının ardından hemen savaşa girmek yerine 1917 yılına kadar tarafsız kalmaya devam etmiştir. Lusitania'nın batırılması, denizaltı savaşlarının sivil gemilere yönelik saldırılarının bir örneği olarak tarihe geçmiştir. Bu olay, denizaltı savaşlarının sivil hayatlara yönelik tehlikelerini ve sivil gemilere karşı yapılan saldırıların sonuçlarını göstermiştir. Aynı zamanda, uluslararası hukuk ve denizcilik kurallarının tartışılmasına ve yeniden düzenlenmesine yol açmıştır [6, 7, 8, 9, 10].

1.2 DSH'IN ÖNEMİ

Önceki maddelerde de görüldüğü üzere denizaltılar dost kuvvetlerin elindeki sivil ve savaş gemilerine karşı oldukça etkili silahlardır. Bu nedenle söz konusu gemilere karşı etkin bir savunma kabiliyetine sahip olunması hayati öneme sahiptir. Bu nedenle çalışmanın bu bölümünde etkin bir DSH kabiliyetine sahip olmanın önemi ele alınacaktır.

Deniz taşımacılığı, dünya ticaretinin büyük bir kısmını oluşturur. Denizaltı saldırıları, deniz taşımacılığını ve ticaret yollarını tehdit ederek ekonomik kayıplara yol açabilir. Ayrıca denizaltı saldırıları, dost unsurların deniz harp gücünü zayıflatır ve savaşta stratejik üstünlüğün kaybedilmesine neden olur. Denizaltılar, sualtında operasyon yapabilme yetenekleri sayesinde düşman gemilerine saldırıda

bulunabilirken, aynı zamanda düşmanın saldırılarından da kaçınabilirler. Bu, düşmanın denizaltıları tespit etme ve hedef alma zorluğu yaşamasına neden olur. Denizaltılar, sürpriz bir saldırı gerçekleştirme yetenekleri sayesinde düşman gemilerine büyük bir tehdit oluştururlar.

DSH, ticaret yollarının güvenliğini sağlamayı ve düşman denizaltılarını algılama, takip etme ve etkisiz hale getirme yeteneklerini geliştirmeyi amaçlar. DSH, modern deniz savaşlarında stratejik bir öneme sahip olan bir savunma taktiğidir.

Etkin bir Denizaltı Savunma Harbi yapısına sahip olunması ile;

- Stratejik dengenin korunması sağlanacaktır. *(Denizaltılar, stratejik füze taşıma kabiliyetine sahip olabilir ve nükleer caydırıcılığın bir parçası olarak kullanılabilir. Denizaltı savunma harbi, bu stratejik dengeyi korumak için oldukça önemlidir.)*
- Denizaltı tehditlerini etkisiz hale getirerek deniz güvenliğini sağlanacaktır.
- Düşmanın denizaltılar ile yapılacak tespit/teşhis/istihbarat avantajı ve imha etme yeteneklerini sınırlandırılabilir. *(Denizaltılar, gizliliklerini koruyarak düşman gemi ve denizaltılarına karşı sürpriz saldırılar yapabilir.)*
- Sahil güvenliğinin sağlanmasına destek oluşturacaktır. *(Denizaltı saldırıları, sahil bölgelerindeki kritik altyapıları, limanları ve kıyı şeridini tehdit edebilir.)*

1.3 LİTERATÜR TARAMASI

Önceki bölümde de anlatıldığı üzere etkin bir DSH yapısına sahip olmak oldukça önemlidir. Bu kapsamda literatürde de bu ve benzer konuları ele alan çalışmalar bulunmaktadır. Söz konusu çalışmalardan bazılarına ilişkin detaylar Tablo 1’de sunulmuştur.

Tablo 1: Literatürde Yer Alan Bazı Benzer Çalışmalar

Çalışma Adı	Kapsam	Kullanılan Model (ler)
Game-Theoretic Anti-Submarine Warfare Mission Planner (heuristic-based, fully Excel capable) [12]	Denizaltı Savunma Harbi Planlaması	Linear Programming Mixed Integer (Linear) Programming
Operational Planning for Theater Anti-Submarine Warfare [13]	Denizaltı Savunma Harbi Planlaması	Navy Mission Planner (NMP)

Tablo 1'in devamı

A Genetic Algorithm Based Anti-Submarine Warfare Simulator [14]	Denizaltı Savunma Harbi	Genetik Algoritma
Anti-Submarine Warfare Search Models [15]	Denizaltı Savunma Harbi	Two Speeds model Three Rays Model Five Rays Model
A Survey of Underwater Acoustic Target Recognition Methods Based on Machine Learning [16]	Sualtı Hedeflerinin Tanınması	Artificial Neural Networks (ANN) Support Vector Machines (SVM) Decision Trees K-Nearest Neighbors Random Forests
A Deep Learning Approach to Dead-Reckoning Navigation for Autonomous Underwater Vehicles With Limited Sensor Payloads [17]	Otonom Sualtı Araçları Navigasyon Sistemleri	Convolutional Neural Networks (CNN) Long Short Term Memory (LSTM)
A Novel Deep Learning Method for Underwater Target Recognition Based on Res-Dense Convolutional Neural Network with Attention Mechanism [18]	Sualtı Hedef Tanıma	CNN
A Selected Review on Reinforcement Learning Based Control for Autonomous Underwater Vehicles [18]	Sürü Halinde Çalışan Otonom Sualtı Araçları	Q-Learning Deep Q Networks Proximal Policy Optimization (PPO) Deep Deterministic Policy Gradient (DDPG)
A Review of Underwater Mine Detection and Classification in Sonar Imagery [19]	Sualtı Hedef Tanıma	CNN Global Average Pooling (GAP) Gabor neural network (GNN)
Underwater Acoustic Research Trends with Machine Learning: General Background [21]	Sualtı Akustiği	SVM ANN K-Means
Navigation of Autonomous Vehicles using Reinforcement Learning with Generalized Advantage Estimation [21]	Otonom Araçlar	Pekiştirmeli Öğrenme

Tablo 1'de yer alan çalışmalar incelendiğinde;

- Game-Theoretic Anti-Submarine Warfare Mission Planner isimli çalışmada doğrusal programlama yöntemi kullanılarak yüksek değerlikli bir platformun tehdit denizaltıdan korunmasına yönelik bir hesaplama yapılmaktadır. Tehdit denizaltıların tamamının imha edilmesine yönelik bir hesaplama yoktur.

- Operational Planning for Theater Anti Submarine Warfare isimli çalışmada Navy Mission Planner isimli model kullanılarak batı pasifikte bir DSH planı oluşturulmuştur. Ancak planlamada hedef imhası dikkate alınmamaktadır. Hedef tespitine yönelik bir planlama söz konusudur.

- A Genetic Algorithm Based Anti Submarine Warfare Simulator isimli çalışmada halihazırda kullanılan bir simülatördeki algoritmanın geliştirilmesine yönelik bir inceleme yapılmıştır. Çevresel model ve denizaltıyı yöneten algoritma yeniden tasarlanmıştır. DSH planlamasına ilişkin bir hesaplama yoktur.

- Anti Submarine Warfare Search Model isimli çalışmada bir MH60 helikopterin DSH görevlerinde nasıl etkinlikle kullanılacağı incelenmiştir. Helikopter olası bir denizaltı hedefinin varlığını tespit etmek, takip etmek, ele geçirmek ve saldırmak için görevlendirilmiş ve bu kapsamda planlamalar yapılmıştır. Bu çalışmada tek bir platform tipi için bir planlama söz konusudur. Genel DSH planlaması yoktur.

- A Survey of Underwater Acoustic Target Recognition Method Based on Machine Learning isimli çalışmada sualtı akustik hedef tanıma yöntemleri incelenmektedir. DSH planlaması yoktur.

- A Deep Learning Approach to Dead-Reckognition Navigation for Autonomus Underwater Vehicles with Limited Sensor Payloads isimli çalışmada makine öğrenme algoritmaları kullanılarak otonom sualtı araçlarının navigasyon bilgileri tahmin edilmeye çalışılmıştır. DSH planlaması yoktur.

- A Novel Deep Learning Method for Underwater Target Recognition Based on Res-Dense Convolutional Neural Network with Attention Mechanism isimli çalışmada sualtı hedef tanıma maksatlı bir model geliştirilmiştir. DSH planlaması yoktur.

- A Selected Review on Reinforcement Learning Based Control for Autonomous Underwater Vehicles isimli çalışmada otonom sualtı araçlarının komutasında pekiştirmeli öğrenme algoritmalarının kullanımını ele almaktadır. DSH planlaması yoktur.

- A Review of Underwater Mine Detection and Classification in Sonar Imagery isimli çalışmada bilgisayar destekli sınıflandırma ve otomatik hedef tanıma algoritmaları üzerine çalışma yapılmıştır. Sualtı mayınlara yönelik bir çalışmadır. DSH planlaması yoktur.

- Underwater Acosutic Resarch Trends with Machine Learning General Background isimli çalışmada sualtı akustiğinde makine öğrenimi tekniklerinin genel teorik arka planı tanıtılmaktadır. DSH planlaması yoktur.

- Navigation of Autonomous Vehicles using Reinorement Learning with Generalized Advantege Estimation isimli çalışmada otonom araç navigasyonlarında pekiştirmeli öğrenme algoritmalarının kullanımına yönelik bir çalışma yapılmıştır. DSH planlaması yoktur.

Önceki maddelerde de açıklandığı üzere söz konusu çalışmaların daha çok hedef tanıma ve otonom araçlar üzerinde yoğunlaştığı, bizim çalışmamızdaki gibi DSH planlamalarında yapay zekayı kullanan bir yaklaşımın bulunmadığı görülmektedir. Bu nedenle bu çalışmanın alanında ilk ve öncü bir değere sahip olacağı değerlendirilmektedir.

1.4 PROBLEMİN TANIMLANMASI

Çalışmanın buraya kadar olan bölümünde denizaltılar ve DSH'nin öneminden bahsedilmiştir. Görüldüğü üzere etkin bir DSH yapısına sahip olmak oldukça önemlidir. Bu nedenle bu çalışmada etkin bir DSH yapısına sahip olunmasına yönelik bir planlama için hesaplamalar yapılmıştır. Bu kapsamda iki farklı problem ele alınmış ve bu problemlerin çözümü bulunmaya çalışılmıştır.

1.4.1 Problem 1

Etkin bir DSH yapısına sahip olabilmek için bu kapsamda kullanabileceğimiz platformların özellikleri ve miktarı oldukça önemlidir. Bu nedenle bu çalışmada ilk olarak düşman kuvvetinin elindeki denizaltıların tamamen imha edilebilmesi için DSH kapsamında kullanılacak belirli özelliklere sahip platformların her birinden kaçar adet sahip olunması gerektiği belirlenmeye çalışılmıştır.

1.4.2 Problem 2

Çalışmanın çözüm aradığı bir diğer problem ise ilk problemde belirlenecek sahip olunması gereken platformlarda hangi mühimmattan kaçar adet bulunması gerektiği çalışılmıştır.

1.5 ÇALIŞMA YÖNTEMİ

Önceki maddelerde açıklanan problemlerin çözülebilmesi için 4 adımdan oluşan bir hesaplama yöntemi uygulanmıştır. Bu kapsamda ilk olarak;

- Çalışmada kullanılacak sentetik veriler üretilmiş,
- Makine öğrenme algoritmaları ile tehdit denizaltıların imha edilebilmesi için atılması gereken mühimmat sayıları hesaplanmış,
- Düşman kuvvete ait denizaltıların tespit edilme olasılıkları belirlenmiş,
- Pekiştirmeli Öğrenme algoritmalarıyla optimal harekât tarzı hesaplanmıştır.

Bahse konu hesaplama adımlarına ilişkin detaylar müteakip maddelerde açıklanmış, çalışmanın genel yapısına ilişkin özet bilgi EK-1’de sunulmuştur.

BÖLÜM II

VERİ TANIMI VE ÖN İŞLEME

2.1 VERİLERİN TANIMLANMASI

Bir DSH'ta hareketin başarısına etki eden birden çok parametre bulunmaktadır. Bahse konu parametreler genel başlıklar altında toplanacak olursa:

- Dost Kuvvet Verileri,
- Düşman Kuvvet Verileri,
- Çevresel Veriler olarak belirlenebilir [22, 23, 24].

Bahse konu veriler milli güvenliğimiz açısından gizlilik derecelerine sahip olduklarından bu çalışmada tamamen sentetik veriler kullanılarak hesaplamalar yapılmıştır. Bu nedenle çalışma kapsamında ilk olarak yukarıdaki parametreleri içeren detayları müteakip maddelerde açıklanmış 220 satırlık kaynak veri setleri (dost/düşman) oluşturulmuştur.

2.1.1 Dost Kuvvet Verileri

Deniz Kuvvetleri Komutanlığı envanterinde yer alan tüm platformların DSH kabiliyeti bulunmamaktadır. Bu nedenle çalışmada dost kuvvet verilerinde sadece DSH yeteneği bulunan platformlar dikkate alınmış olup, çalışma kapsamında hesaplama dâhil edilen platform bilgileri Tablo 2'de sunulmuştur. Tablo 2'de yer alan verilere ilişkin açıklamalar aşağıda sunulmuştur.

2.1.1.1 Parametre Açıklamaları

Sonar menzili; tehdit denizaltının tespit edilmesi ve angajman gerçekleştirilmesi açısından önemli bir parametredir. Sonar menzili arttıkça platformun tehdit denizaltıyı tespit etme olasılığı da artacaktır [22, 23, 24].

Sonar yaşı; kullanılan sonarın sahip olduğu teknoloji ile ilgilidir. Bu nedenle sonar yaşı büyük olan platformlar daha eski teknolojiye sahip sonara sahip olacaktır. Bu durum tehdit denizaltının tespit edilme durumunu azaltacaktır [22, 23, 24].

Sonar frekansı; suyun altında tespit işlemi gönderilen ses dalgaları ile yapılmaktadır. Bu nedenle sonarların sahip olduğu frekans büyüklüğü sonar ile yapılacak tespit durumunu da değiştirecektir. Bu nedenle sonarların frekansları arttıkça tespit etme durumu da artacaktır [22, 23, 24].

Tonaj bilgisi tehdit denizaltılar tarafından atılacak silah miktarlarının hesaplanmasında kullanılacaktır. Angajman gerçekleştirilen platformun tonajı arttıkça imha etmek için gerekecek silah sayısı da artacaktır.

Uzunluk bilgisi tehdit denizaltılar tarafından atılacak silah miktarlarının hesaplanmasında kullanılacaktır. Angajman gerçekleştirilen platformun uzunluğu arttıkça platformun tonajı da artacak bu durumun imha etmek için gerekecek silah sayısını da artıracaktır.

Platformun sürati tehdit denizaltının tespit olasılığının hesaplanmasında kullanılacak bir parametredir. Söz konusu olasılık hesabına ilişkin hesaplama detayları müteakip maddelerde açıklanacaktır [26].

Maliyet bilgisi çalışmanın pekiştirmeli öğrenme ile yapılacak hesaplamalarda ceza puanının belirlenmesinde kullanılacaktır. Harekât tarzı belirlenirken dost platform imha edilmesi durumunda platformun maliyeti kadar ceza puanı alacaktır. Bu nedenle tehdit denizaltının imhasında daha düşük maliyete sahip platformlarca angajman gerçekleştirilmeye çalışılacak bu durumun yüksek değerli platformların korunmasını sağlayacaktır. Bu kapsamda hesaplamada kullanılacak platform maliyetleri açık kaynaklarda yer alan değerlere göre belirlenmiştir [27, 28, 29, 30, 31].

Silah maliyeti de çalışmanın pekiştirmeli öğrenme ile yapılacak hesaplamalarda ceza puanının belirlenmesinde kullanılacaktır. Harekât tarzı belirlenirken dost platform tarafından atılacak silahların maliyeti kadar ceza puanı alacaktır. Bu nedenle tehdit denizaltının imhasında daha düşük maliyete sahip silahlar ile angajman gerçekleştirilmeye çalışılacak, bu durum daha maliyet etkin bir DSH planlaması yapılmasını sağlayacaktır. Bu kapsamda hesaplamada kullanılacak silah maliyetleri açık kaynaklarda yer alan değerlere göre belirlenmiştir [31].

2.1.2 Düşman Kuvvet Verileri

Çalışma kapsamında kullanılan dost unsurlar tarafından angajman gerçekleştirilecek düşman kuvvet denizaltılarına ilişkin verileri ifade etmektedir. Bu kapsamda düşman kuvvetin 5 tip denizaltıdan oluştuğu varsayılmıştır. Çalışmada

kullanılan düşman kuvvet verilerine ilişkin örnek gösterim detaylar Tablo 3’te sunulmuştur. Düşman kuvvet verileri de dost kuvvet verileri ile benzer parametrelerden oluşturulmuştur. Ancak anılan verilere ilave olarak sualtında kalma süresi parametresi eklenmiştir. Sualtında kalma süresi tehdit denizaltının tespit olasılığının hesaplanmasında kullanılacak bir parametredir. Söz konusu olasılık hesabına ilişkin hesaplama detayları müteakip maddelerde açıklanacaktır. Anılan parametre dışındaki diğer parametreler dost kuvvet verilerindeki parametreler ile aynı amaçla kullanılacaktır. Sadece taraf olarak değişiklik olacaktır. Örneğin maliyet bilgisi yine çalışmanın pekiştirmeli öğrenme ile yapılacak hesaplamalarında ceza puanının belirlenmesinde kullanılacaktır. Çalışmada sadece dost kuvvetin değil aynı zamanda düşman kuvvetin de harekât tarzı belirlenmektedir. Bu nedenle düşman kuvvetin harekât tarzı belirlenirken; düşman kuvvete ait bir denizaltının imha edilmesi durumunda denizaltının maliyeti kadar ceza puanı alacaktır. Bu nedenle düşman kuvvet dost kuvvete ait platformları imha etmek için daha düşük maliyete sahip denizaltılarca angajman gerçekleştirilmeye çalışılacak bu durumun yüksek değerli denizaltıların korunmasını sağlayacaktır.

2.1.3 Çevresel Veriler

DSH’ta en önemli faktör tehdit denizaltının tespit edilmesidir. Sualtındaki bir platform/cismin tespit edilmesinde ise en önemli etken sesin iletilmesidir [22, 23, 24]. Bu nedenle DSH kapsamında yapılacak hesaplama/planlamalarda sesin sualtında iletimini etkileyen tüm faktörlerin göz önünde bulundurulması gerekmektedir. Bu kapsamda sesin sualtında iletilmesini etkileyen ve çalışmada dikkate alınan faktörler Tablo 4’te sunulmuştur. Bu çalışmada Ege Denizinde icra edilecek bir DSH harekâtı için analiz yapılmıştır. Bu nedenle bu bölgedeki çevresel veriler kullanılmıştır. Ancak mevsimsel değişimler anılan verilerin de değişmesine sebep olacaktır. Bu nedenle çalışmada en kötü senaryoya yönelik planlama yapılabilmesi için (deniz suyu sıcaklığı düştüğünde hedef tespiti daha zor olacaktır.) kış mevsiminin (Aralık, Ocak, Şubat dönemi) çevresel verileri dikkate alınarak hesaplamalar yapılmıştır. Tablo 4’te yer alan verilere ilişkin açıklamalar aşağıda sunulmuştur.

2.1.3.1 Parametre Açıklamaları

Sesin su içerisinde yayılmasını etkileyen en önemli faktör sıcaklıktır. Sıcaklık arttıkça ses hızı da artış gösterecektir. Ses hızının artması hedefin tespit olasılığını da

artıracaktır [22, 23, 24]. Bu kapsamda kaynak veri setlerinde ortalaması 13,13, standart sapması 5 olan normal dağılıma uygun 220 adet sıcaklık değeri oluşturulmuştur [33].

Deniz suyu tuzluluk oranının binde bir artışı ses hızında 1.3 m/s'lik artışa sebep olmaktadır. Ses hızının artması tespit olasılığını artırdığından deniz suyu tuzluluk oranı arttıkça hedefin tespit olasılığı da artacaktır [22, 23, 24]. Bu kapsamda kaynak veri setlerinde ortalaması 36,22, standart sapması 4 olan normal dağılıma uygun 220 adet deniz suyu tuzluluk değeri oluşturulmuştur [34].

Deniz suyu yoğunluğu tuzluluk ve sıcaklık parametrelerine göre değişiklik gösterir. Yoğunluktaki artış ses hızını da artırır. Ses hızının artması tespit olasılığını artırdığından deniz suyu yoğunluğu arttıkça hedefin tespit olasılığı da artacaktır [22, 23, 24]. Bu kapsamda kaynak veri setlerinde ortalaması 26,05, standart sapması 1,5 olan normal dağılıma uygun 220 adet deniz suyu yoğunluğu değeri oluşturulmuştur [35].

Sualtındaki platform/cisimlerin tespitinde gürültü değerleri de oldukça büyük öneme sahiptir. Ortamda yer alan gürültü değerleri sesin iletilmesinde engel oluşturabileceklerdir. Bu nedenle ortamda yer alan gürültü değerlerinin de hesaplamada dikkate alınması gerekmektedir. Bu kapsamda çalışmada, denizaltında oluşabilecek detayları müteakip maddelerde açıklanan gürültü değerleri dikkate alınmıştır.

Deniz trafiği gürültüsü değeri deniz trafiğini oluşturan gemilerin pervane ve makine gürültüleri ile yükleme, boşaltma, yedekleme, deniz sanayi tesislerindeki makinelerin yol açtığı gürültülerdir. Bu gürültüler rüzgâr ve hava şartlarına bağlı değildir [22, 23, 24]. Bu kapsamda kaynak veri setlerinde ortalaması 62, standart sapması 5 olan normal dağılıma uygun 220 adet deniz trafiği gürültüsü değeri oluşturulmuştur [35].

Çevre gürültüsü kendi gürültümüz ile bazı özel gürültü kaynağı gemilerin yaydığı gürültü dışında kalan denizin kendi fon gürültüsüdür. Çevre gürültüsü bulunan ortama göre değişiklik gösterir. Çevre gürültüsü satih gürültüsü (su yüzeyindeki dalga, rüzgâr ve yağış gürültüleri), biyolojik gürültüler (deniz canlılarından kaynaklanan gürültüler) ve deniz suyu türbülansı (deniz içerisinde oluşan tüm akıntı ve türbülanslar) sebebiyle oluşan gürültülerdir. Bu kapsamda kaynak veri setlerinde ortalaması 30, standart sapması 5 olan normal dağılıma uygun 220 adet çevre gürültüsü değeri oluşturulmuştur [36].

Platformun kendi gürültüsü; sadece kendi gemimiz tarafından oluşturulan gürültü değeridir. Kendi gürültümüz hedeften gelen ekoyu maskeleyiği için tespit olasılıklarına da olumsuz etki eder. Bu nedenle hedef tespitinde dikkate alınması gereken önemli bir husustur. Bu kapsamda kaynak veri setlerinde ortalaması 27, standart sapması 5 olan normal dağılıma uygun 220 adet platform gürültüsü değeri oluşturulmuştur [37].

Dünya üzerinde sürekli olarak sismik aktiviteler oluşmaktadır. Bu durum deniz altında düşük frekanslı da olsa bir gürültüye sebebiyet vermektedir. Sismik gürültüler hedeften gelen ekoyu maskeleyiği için tespit olasılıklarına da olumsuz etki eder. Bu nedenle hedef tespitinde dikkate alınması gereken önemli bir husustur. Bu kapsamda kaynak veri setlerinde ortalaması 0,9, standart sapması 0,5 olan normal dağılıma uygun 220 adet sismik gürültü değeri oluşturulmuştur [22, 23, 24].

Tüm bu veriler ile oluşturulan kaynak veri setinin örnek gösterimi Tablo 5'te sunulmuştur.

Tablo 2: Çalışmada Kullanılan Dost Kuvvet Verileri

Platform Tipi	Platform Miktarı (adet)	Sonar Menzili (Deniz mili)	Sonar Yaşı (yıl)	Sonar Frekansı (Hz)	Tonajı (ton)	Uzunluk (metre)	Sürati (knot)	Maliyeti (milyon \$)	Silah Tipi	Silah Maliyeti (milyon \$)
Fırkateyn Tip 1	6	46	45	6,7	4000	135	30	836	T1	1,8
									T2	2
Fırkateyn Tip 2	3	67	37	8,2	3000	110	27	1169	T2	2
									T3	2,1
Fırkateyn Tip 3	3	48	24	7	3100	118	32	885	T1	1,8
									T3	2,1
Korvet	3	44	8	7	2400	100	30	1126	T2	2
									T3	2,1
Helikopter Tip 1	8	56	35	8	2	14	130	5	T2	2
									T3	2,1
Helikopter Tip 2	8	54	30	7,6	2	14	130	6	T1	1,8
									T2	2
Deniz Karakol Uçağı	4	49	25	7,3	10	50	240	57	T1	1,8
									T3	2,1
Denizaltı Tip 1	3	54	35	7,7	1400	65	22	1775	T2	2
									T3	2,1
Denizaltı Tip 2	3	53	25	6,7	1800	75	22	1483	T1	1,8
									T2	2
Denizaltı Tip 3	3	66	20	6,8	1500	70	22	1359	T1	1,8
									T3	2,1
Denizaltı Tip 4	3	48	2	7,5	1200	60	22	1979	T1	1,8
									T2	2

Tablo 3: Çalışmada Kullanılan Düşman Kuvvet Verileri

Platform Tipi	Platform Miktarı (adet)	Sonar Menzili (Deniz mili)	Sonar Yaşı (yıl)	Sonar Frekansı (Hz)	Tonajı (ton)	Uzunluk (metre)	Sürati (knot)	Maliyeti (milyon \$)	Sualtında Kalma Süresi (saat)	Silah Tipi	Silah Maliyeti (milyon \$)
Tehdit Denizaltı Tip 1	4	40	35	7	1100	60	18	1523	15	T4	2
										T5	2,2
Tehdit Denizaltı Tip 2	5	55	27	7,4	1400	65	20	1179	20	T5	2,2
										T6	1,7
Tehdit Denizaltı Tip 3	7	43	18	8	1800	75	23	1374	18	T4	2
										T6	1,7
Tehdit Denizaltı Tip 4	8	60	12	8,2	1200	60	27	1149	23	T4	2
										T5	2,2
Tehdit Denizaltı Tip 5	6	57	4	8,3	1500	70	30	1269	28	T5	2,2
										T6	1,7

Tablo 4: Çalışmada Kullanılan Çevresel Veriler

Parametre Adı	Deniz Suyu Sıcaklığı (C ⁰)	Deniz Suyu Tuzluluk Oranı (%)	Deniz Suyu Yoğunluğu (kg/m ³)	Gürültü Verileri			
				Deniz Trafığı Gürültüsü (dBA)	Çevresel Gürültüler (dBA)	Platformun Kendi Gürültüsü (dBA)	Sismik Gürültüler (dBA)
Değer Aralığı	10,13-17,25	32,15-38,86	25,5-27,3	57,8-68,1	25,3-34,7	21,8-30,2	0,5-1,4

Tablo 5: Dost Kuvvet Angajman Veri Tabanı Kaynak Veri Seti Örnek Gösterimi

Sonar Menzili	Sonar Yaşı	Sonar Frekansı	Deniz Suyu Sıcaklığı	Deniz Suyu Tuzluluk Oranı	Deniz Suyu Yoğunluğu	Deniz Trafik Gürültüsü	Çevresel Gürültüler	Platformun Kendi Gürültüsü	Sismik Gürültüler	Hedef Tonajı	Hedef Uzunluğu	Atılan Mühimmat Sayısı		
												T1	T2	T3
56	35	8.00	13.35	37.49	25.33	66.01	26.74	26.55	1.19	0	0	0	2	2
67	37	8.20	18.45	32.07	24.09	62.29	31.86	33.59	0.19	0	0	0	3	0
44	8	7.00	11.13	30.30	26.52	65.62	36.63	19.84	1.56	1400	65	0	3	0
49	25	7.30	7.85	34.17	24.23	63.17	31.18	31.14	0.70	0	0	3	0	2
66	20	6.80	5.67	41.35	27.15	60.96	30.14	25.54	0.96	0	0	3	0	0
53	25	6.70	14.54	35.33	28.76	70.51	34.22	21.04	0.75	0	0	4	0	0
48	24	7.00	15.67	38.72	28.25	60.05	31.65	29.82	1.42	0	0	0	0	3
53	25	6.70	14.72	32.92	27.99	64.98	29.94	16.54	0.73	1400	65	2	3	0
67	37	8.20	23.55	35.46	26.62	67.41	31.82	34.55	0.58	1100	60	0	6	0
48	24	7.00	14.51	41.91	27.32	61.75	30.08	31.13	0.62	1200	60	0	0	3
66	20	6.80	6.12	40.16	26.29	66.17	33.00	24.40	0.71	1800	75	2	0	3
44	8	7.00	11.94	33.69	26.39	59.85	34.93	39.43	0.12	0	0	0	1	2
67	37	8.20	18.45	32.07	24.09	62.29	31.86	33.59	0.19	0	0	0	3	0
49	25	7.30	16.07	31.96	25.62	67.72	39.24	23.89	0.65	1500	70	0	0	3
66	20	6.80	11.08	31.24	27.81	69.35	33.87	27.89	2.44	1500	70	1	0	5
48	2	7.50	18.45	30.05	24.53	56.41	23.02	31.77	0.63	1100	60	2	3	0

2.2 SENTETİK VERİLERİN ÜRETİLMESİ

Çalışmanın önceki bölümünde 220 satırlık kaynak veri setleri oluşturulmuştur. Ancak anılan veri sayısı doğru tahmin yapmak için yeterli olmayacaktır. Bu nedenle çalışmanın bu bölümünde önceki bölümde oluşturulmuş olan kaynak veri setleri kullanılarak veri sayısı artırılacaktır. Bu kapsamda detayları müteakip maddelere açıklanmış olan yöntemler kullanılarak her bir yöntem ile veri sayıları 10000 yapılmıştır.

2.2.1 Sentetik Veri Üretme Tekniklerinin İncelenmesi

Literatürde birçok sentetik veri üretme tekniği bulunmaktadır. Bu çalışmada literatürde daha çok kullanılan 8 yöntem ile veri artırımı gerçekleştirilmiştir. Bu kapsamda;

- Sentetik Azınlık Aşırı Örnekleme Tekniği (Synthetic Minority Over Sampling Technique SMOTE) Yöntemi,
- Borderline-SMOTE Yöntemi
- Uyarlamalı Sentetik Örnekleme (Adaptive Synthetic Sampling, ADASYN) Yöntemi
- Rastgele Aşırı Örnekleme Örnekleri (Random Over Sampling Examples, ROSE) Yöntemi
- Trim-SMOTE Yöntemi,
- Çoğunluk Ağırlıklı Azınlık Aşırı Örnekleme Tekniği (Majority Weighted Minority Oversampling Technique, MWMOTE) Yöntemi,
- Hiyerarşik Aglomeratif Kümeleme (Hierarchical Agglomerative Clustering, HAC) Yöntemi,
- Karışım Yoğunluğu Aşırı Örnekleme (Mixture Density Oversampling, MDO) Yöntemi kullanılarak her bir yöntem ile veri sayıları 10000 olacak şekilde veri artırımları yapılmıştır. Söz konusu yöntemlere ilişkin detaylar müteakip maddelerde, olumlu ve olumsuz yönleri Tablo 6'da sunulmuştur.

2.2.1.1 SMOTE Yöntemi

SMOTE yöntemi, azınlık sınıflara sahip veri kümelerinde, özellikle sınıf dengesizliği problemiyle karşı karşıya kalan makine öğrenmesi algoritmalarında sıklıkla kullanılan bir veri artırma tekniğidir. Bu yöntem, azınlık sınıfındaki

örneklerin sayısını arttırmak için sentetik veriler üretir. SMOTE, veri örneklerinin sentetik örneklerle çoğaltılması için kullanılır. Bu yöntemde, azınlık sınıfındaki örneklerin her biri seçilir ve bunların her biri ile rastgele seçilen bir diğer örnek arasındaki farklar hesaplanır. Ardından, bu farkların bir çarpanı ile çarpılır ve seçilen örnek ile çarpılmış farklar toplanarak yeni bir örnek oluşturulur. SMOTE yöntemi, yeni verilerin üretilmesi için özellikle azınlık sınıfındaki örnekler arasında etkileşim kurar. Bu yöntem, verilerin dağılımını değiştirmez, yalnızca verilerin arasındaki mesafeyi artırarak örneklerin daha iyi sınıflandırılmasına yardımcı olur. SMOTE yöntemi, veri örneklerinin sayısını arttırdığı için, aynı zamanda modelin overfitting yapmasını da önler. Ancak, veri kümelerinin büyümesi, modelin daha yavaş çalışmasına neden olabilir. SMOTE yöntemi, sınıf dengesizliği problemiyle karşı karşıya kalan pek çok algoritma için bir çözüm sunar. Ancak, bu yöntemle elde edilen sentetik veriler gerçek verilerle aynı olmadığı için, dikkatli bir şekilde kullanılmalıdır ve sonuçlar her zaman doğru olmayabilir [39].

2.2.1.2 Borderline-SMOTE Yöntemi

Borderline-SMOTE yöntemi, SMOTE yöntemi ile benzer bir şekilde, sınıf dengesizliği problemini çözmek için kullanılan bir veri artırma tekniğidir. Ancak, Borderline-SMOTE yöntemi, SMOTE yöntemine kıyasla daha az sayıda sentetik veri üretir ve sentetik verilerin daha doğru olmasını sağlar. Borderline-SMOTE yöntemi, yalnızca sınıf sınırına yakın olan örnekleri seçerek sentetik veri üretir. Bu sayede, veri kümelerinin dağılımında büyük bir değişiklik oluşmaz ve sentetik veriler gerçek verilere daha yakın olur. Borderline-SMOTE yöntemi, iki aşamada gerçekleştirilir. İlk aşamada, veri kümesindeki her bir azınlık sınıfı örneği analiz edilir ve her bir örneğin sınıfına olan uzaklığı hesaplanır. Uzaklık ölçütü, örneğin en yakın komşularının oranı olarak belirlenebilir. Bu aşamada, azınlık sınıfındaki örnekler, sınıf sınırına yakın olanlar ve uzak olanlar olarak iki gruba ayrılır. İkinci aşamada, sentetik veriler üretilir. Sınıf sınırına yakın olan örnekler, rastgele seçilen birkaç örnekle birleştirilerek sentetik veriler oluşturulur. Ancak, uzak olan örneklerin sentetik verilerle birleştirilmesi engellenir. Bu sayede, veri kümesinin dağılımı korunur ve sentetik veriler gerçek verilere daha uygun hale gelir. Borderline-SMOTE yöntemi, sınıf dengesizliği problemiyle karşı karşıya kalan pek çok algoritma için bir çözüm sunar. Ancak, bu yöntemle elde edilen sentetik veriler de SMOTE yöntemi

gibi gerçek verilerle aynı olmadığı için, dikkatli bir şekilde kullanılmalıdır ve sonuçlar her zaman doğru olmayabilir [39].

2.2.1.3 ADASYN Yöntemi

ADASYN yöntemi, sınıf dengesizliği problemini çözmek için kullanılan bir veri artırma tekniğidir. Bu yöntem, SMOTE yöntemi gibi azınlık sınıfındaki örneklerin sayısını arttırmak için sentetik veri üretir, ancak SMOTE yönteminden farklı olarak, her bir azınlık sınıfı örneği için farklı bir sentetik veri oranı kullanır. ADASYN yöntemi, azınlık sınıfındaki her bir örnek için, en yakın komşularının sayısına dayalı bir ağırlık değeri hesaplar. Bu ağırlık değeri, sentetik veri oranının belirlenmesinde kullanılır. Yani, azınlık sınıfındaki örneklerin sayısına göre farklı sentetik veri oranları belirlenir. Böylece, veri kümesindeki azınlık sınıfının daha az temsil edildiği bölgelerde daha fazla sentetik veri üretilir. ADASYN yöntemi, iki aşamada gerçekleştirilir. İlk aşamada, azınlık sınıfındaki örneklerin en yakın komşuları belirlenir ve ağırlık değerleri hesaplanır. İkinci aşamada, her bir azınlık sınıfı örneği için sentetik veriler üretilir. Sentetik veriler, rastgele seçilen bir azınlık sınıfı örneği ile en yakın komşuları arasındaki farkın bir kısmı kadar uzaklıkta yer alan noktalardan oluşur. ADASYN yöntemi, SMOTE yöntemine kıyasla daha az sayıda sentetik veri üretir ve sentetik verilerin daha doğru olmasını sağlar. Ayrıca, sentetik veriler, azınlık sınıfındaki örneklerin dağılımına daha uygun hale getirilir. Bu sayede, azınlık sınıfındaki örneklerin temsil edilme oranı artar ve sınıf dengesizliği problemi çözülebilir. ADASYN yöntemi, sınıf dengesizliği problemiyle karşı karşıya kalan pek çok algoritma için bir çözüm sunar. Ancak, sentetik veriler gerçek verilerle tam olarak aynı olmadığı için, dikkatli bir şekilde kullanılmalıdır ve sonuçlar her zaman doğru olmayabilir [40].

2.2.1.4 ROSE Yöntemi

ROSE yöntemi, sınıf dengesizliği problemi ile başa çıkmak için kullanılan bir veri artırma tekniğidir. Bu yöntem, azınlık sınıfındaki örneklerin sayısını arttırmak için rastgele seçilen verilerin kopyalanması yoluyla sentetik veriler oluşturur. ROSE yöntemi, veri kümesindeki azınlık sınıfındaki örneklerin sayısını arttırmak için rastgele seçilen örnekleri kopyalar. Kopyalama işlemi, rastgele seçilen bir azınlık sınıfı örneğinin birkaç kopyasının üretilmesi yoluyla gerçekleştirilir. Bu sayede, azınlık sınıfındaki örneklerin sayısı artırılarak sınıf dengesizliği problemi

özlebilir. ROSE yntemi, SMOTE yntemine kıyasla daha basit bir yntemdir. Ayrıca, sentetik veriler oluřturmak iin herhangi bir zel hesaplama yapılmadıđı iin daha hızlıdır. Ancak, sentetik veriler gerek verilerle aynı deđildir, bu nedenle veri kmesindeki azınlık sınıfının zellikleri tam olarak temsil edilemeyebilir. ROSE yntemi, sentetik veri retmek iin rastgele seilen rneklerin kopyalanması yoluyla alıřtıđından, retilen verilerin gerek verilerle aynı dađılıma sahip olması garantisi yoktur. Bu nedenle, ROSE yntemi kullanılırken, veri kmesindeki her bir rneđin nem dzeyini dikkate almak nemlidir. rneđin, veri kmesindeki bazı rnekler diđerlerinden daha nemli olabilir, bu nedenle bu rneklerin kopyalanması daha az tercih edilebilir. ROSE yntemi, sınıf dengesizliđi problemiyle karřı karřıya kalan pek ok algoritma iin bir zm sunar. Ancak, sentetik veriler gerek verilerle tam olarak aynı olmadığı iin, dikkatli bir řekilde kullanılmalıdır ve sonular her zaman dođru olmayabilir [42].

2.2.1.5 Trim-SMOTE Yntemi

Trim-SMOTE, sınıf dengesizliđi problemini zmek iin kullanılan bir veri artırma yntemidir. Bu yntem, SMOTE yntemini temel alır, ancak bu yntemde, sınıf dengesizliđinin azaltılması iin kullanılan sentetik rnekler, SMOTE yntemine kıyasla daha dikkatli bir řekilde oluřturulur. Trim-SMOTE yntemi, nce SMOTE yntemi kullanılarak sentetik rnekler oluřturur. Daha sonra, oluřturulan sentetik rnekler, daha az nemli olan rneklerin kaldırılması ile dzenlenir. Bu yntem, sentetik rneklerin kalitesini arttırır ve daha dengeli bir veri kmesi oluřturulmasına yardımcı olur. Trim-SMOTE yntemi, SMOTE yntemine kıyasla daha yavaş alıřabilir. Ancak, sentetik rneklerin daha kaliteli olması, sonuların daha iyi olmasına yardımcı olur. Trim-SMOTE yntemi, azınlık sınıfındaki rneklerin sayısını nemli lde arttırmayabilir, ancak sentetik rneklerin daha kaliteli olması, sınıf dengesizliđi problemini zmeye yardımcı olabilir. Trim-SMOTE yntemi, veri kmesindeki azınlık sınıfındaki rneklerin sayısını arttırırken, aynı zamanda rneklerin sayısını dzenleyerek daha dengeli bir veri kmesi oluřturulmasına yardımcı olur. Bu yntem, sınıf dengesizliđi problemiyle karřı karřıya olan pek ok makine đrenmesi algoritması iin bir zm sunar. Trim-SMOTE yntemi, sınıf dengesizliđi problemini zmek iin etkili bir yntemdir. Ancak, sentetik rneklerin daha kaliteli olması iin, veri kmesinin zelliklerine dikkat edilmesi nemlidir [42].

2.2.1.6 MWMOTE Yöntemi

MWMOTE yöntemi, sınıf dengesizliği problemini çözmek için kullanılan bir veri artırma yöntemidir. Bu yöntem, azınlık sınıfındaki örneklerin sentetik örneklerle çoğaltılmasını sağlar, ancak aynı zamanda örneklerin önem derecesini de göz önünde bulundurarak, veri artırma işlemini gerçekleştirir. MWMOTE yöntemi, her azınlık sınıfı örneği için bir ağırlık değeri hesaplar. Ağırlık değeri, örneğin azınlık sınıfının ne kadar önemli olduğunu gösterir. Bu ağırlık değeri, azınlık sınıfındaki örneklerin üretilen sentetik örneklerle çoğaltılırken dikkate alınır. MWMOTE yöntemi, azınlık sınıfındaki örneklerin sentetik örneklerle çoğaltılması için kullanılan SMOTE yöntemini temel alır. Ancak MWMOTE yöntemi, azınlık sınıfındaki örneklerin ağırlık değerlerini de dikkate alarak, sentetik örnekleri üretir. Bu sayede, daha az önemli olan azınlık sınıfı örneklerine kıyasla, daha önemli olan azınlık sınıfı örneklerinin daha fazla sentetik örnekle çoğaltılması sağlanır. MWMOTE yöntemi, azınlık sınıfındaki örneklerin sayısını arttırmak için kullanılan diğer veri artırma yöntemlerine kıyasla daha yüksek kaliteli sentetik örnekler üretir. Bu sayede, sınıf dengesizliği problemi daha etkili bir şekilde çözülebilir. MWMOTE yöntemi, sentetik örneklerin üretiminde azınlık sınıfındaki örneklerin ağırlık değerlerini dikkate aldığı için, sınıf dengesizliği problemiyle karşı karşıya olan pek çok makine öğrenmesi algoritması için etkili bir çözüm sunar. Ayrıca, MWMOTE yöntemi, azınlık sınıfındaki örneklerin sayısını arttırırken, aynı zamanda veri kümesindeki sınıf dengesizliğinin daha iyi bir şekilde ele alınmasına yardımcı olur. Sonuç olarak, MWMOTE yöntemi, sınıf dengesizliği problemini çözmek için etkili bir yöntemdir. Azınlık sınıfındaki örneklerin ağırlık değerlerini dikkate alarak, daha yüksek kaliteli sentetik örnekler üretir ve bu sayede, sınıf dengesizliği problemi daha etkili bir şekilde çözülebilir. Bu yöntem, diğer veri artırma yöntemleriyle kıyaslandığında daha iyi sonuçlar verir ve daha az yanlış sınıflandırma yapılmasına yardımcı olur. MWMOTE yöntemi, veri kümesindeki sınıf dengesizliğini azaltmak için kullanılabilecek etkili bir yöntemdir. Ancak, bu yöntemin de bazı dezavantajları vardır. Örneğin, ağırlık değerleri hesaplanırken, özellikle büyük veri kümelerinde, hesaplama maliyeti yüksek olabilir. Ayrıca, MWMOTE yöntemi, azınlık sınıfındaki örneklerin ağırlık değerlerini doğru bir şekilde hesaplayamadığı zaman, yanlış sonuçlar üretebilir. Sonuç olarak, MWMOTE yöntemi, sınıf dengesizliği problemiyle karşı karşıya olan pek çok makine öğrenmesi algoritması için etkili bir çözüm sunar. Ancak, veri

kümesinin özelliklerine ve sınıf dengesizliği oranına bağlı olarak, diğer veri artırma yöntemleri de kullanılabilir [42].

2.2.1.7 HAC Yöntemi

HAC yöntemi, sınıflandırma problemlerinde kullanılan bir veri üretme yöntemidir. Bu yöntem, hiyerarşik bir kümeleme algoritması kullanarak veri kümesindeki benzer örnekleri gruplar halinde birleştirir ve yeni sentetik örnekler üretir. HAC yöntemi, öncelikle veri kümesindeki tüm örneklerin benzerlik matrisi hesaplanarak başlar. Benzerlik matrisi, tüm örneklerin birbirleriyle olan benzerliklerini ölçer ve örneklerin benzerliklerini bir matris şeklinde tutar. Bu benzerlik matrisi, örnekler arasındaki uzaklıkları ölçerek hesaplanır. Daha sonra, bu benzerlik matrisi kullanılarak, örnekler kümeleme algoritmasıyla gruplara ayrılır. Bu işlem sırasında, en yakın örnekler birleştirilerek yeni bir küme oluşturulur. Bu kümeleme işlemi, belirli bir küme sayısına veya birleştirme kriterine ulaşıncaya kadar devam eder. HAC yöntemi, bu kümeleme işlemi sırasında, örneklerin birleştirilmesiyle yeni sentetik örnekler üretir. Bu sentetik örnekler, benzerlik matrisindeki örnekler arasındaki uzaklık ve benzerlik değerlerine göre üretilir. Özellikle azınlık sınıfındaki örneklerin birleştirilmesiyle sentetik örnekler üretilir. HAC yöntemi, sınıflandırma problemlerinde azınlık sınıfındaki örneklerin sayısını artırmak için kullanılabilir. Ancak, HAC yöntemiyle üretilen sentetik örneklerin kalitesi, kümeleme algoritmasının kalitesine ve kullanılan benzerlik matrisine bağlıdır. Sonuç olarak, HAC yöntemi, sınıf dengesizliği problemiyle karşı karşıya olan sınıflandırma problemleri için etkili bir çözüm sunar. Ancak, veri kümesinin özelliklerine ve sınıf dengesizliği oranına bağlı olarak, diğer veri artırma yöntemleri de kullanılabilir [42].

2.2.1.8 MDO Yöntemi

MDO yöntemi, sınıf dengesizliği problemiyle karşı karşıya olan sınıflandırma problemleri için bir veri artırma yöntemidir. Bu yöntem, sınırlı sayıda azınlık sınıfı örneğinden yararlanarak, özellikle az sayıda örneği olan azınlık sınıfının dağılımını modellenerek yeni sentetik örnekler üretir. MDO yöntemi, öncelikle azınlık sınıfındaki örneklerin özelliklerini ve dağılımlarını bir Gaussian karışım modeli (Gaussian Mixture Model - GMM) ile modeller. Bu model, azınlık sınıfındaki örneklerin dağılımını ölçer ve bu dağılım bilgisi, sentetik örneklerin üretilmesinde

kullanılır. Daha sonra, MDO yöntemi, azınlık sınıfındaki örneklerin modele uygun şekilde çekilerek ve farklı ağırlıklar verilerek sentetik örnekler üretir. Bu sentetik örnekler, azınlık sınıfındaki örneklerin dağılımına uygun şekilde üretilir ve azınlık sınıfının daha iyi temsil edilmesine katkıda bulunur. MDO yöntemi, GMM modelinin parametrelerini belirlemek için EM (Expectation-Maximization) algoritmasını kullanır. EM algoritması, verilerin karışık bir dağılım modeli ile modellendiği durumlarda kullanılan bir öğrenme algoritmasıdır. Bu algoritma, verilerin gizli sınıflandırması ve parametre tahmini işlemini gerçekleştirir. MDO yöntemi, sentetik örneklerin kalitesini artırmak için birçok farklı GMM modeli kullanabilir. Örneğin, farklı GMM modelleri, farklı sayıda bileşen içerebilir veya farklı koşullar altında kullanılabilir. Sonuç olarak, MDO yöntemi, az sayıda örneği olan azınlık sınıfının dağılımını modeller ve sentetik örneklerin üretiminde bu modeli kullanarak daha iyi bir azınlık sınıfı temsili sağlar. Ancak, bu yöntemin etkililiği, özellikle veri kümesinin boyutuna ve sınıf dengesizliği oranına bağlıdır. Ayrıca, modelin doğru parametrelerle eğitilmesi ve sentetik örneklerin kalitesinin doğru bir şekilde ölçülmesi de önemlidir [42].

Tablo 6: Sentetik Veri Üretme Tekniklerinin Olumlu ve Olumsuz Yönleri

Sentetik Veri Üretme Tekniği	Olumlu Yönleri	Olumsuz Yönleri
SMOTE	SMOTE yöntemi, az sayıdaki sınıfın örnek sayısını artırarak sınıf dengesizliği sorununa etkili bir çözüm sunar. Bu sayede, modelin az örneklenen sınıfları tanıması ve doğru sınıflandırma yapması sağlanır. SMOTE yöntemi, sentetik örnekler oluşturarak gerçek dünya örneklerine benzer veriler üretebilir. Bu, modelin daha gerçekçi bir şekilde eğitilmesini sağlayarak daha iyi performans elde edilmesine yardımcı olur. SMOTE yöntemi, overfitting riskini azaltır. Bu, modelin eğitim verilerine aşırı uyum sağlamasını engelleyerek genelleştirilebilir bir model elde edilmesine yardımcı olur.	SMOTE yöntemi, bazı durumlarda aşırı örnekleme yaparak veri setindeki çeşitliliği azaltabilir. Bu durum, modelin gerçek dünya verilerinde başarısız olmasına neden olabilir. SMOTE yöntemi, sentetik örnekler oluştururken gerçek örneklerin doğal dağılımını değiştirebilir. Bu, modelin gerçek dünya verilerinde hatalı sınıflandırma yapmasına neden olabilir. SMOTE yöntemi, az sayıdaki sınıfın örneklerine dayandığı için, bu sınıfların örnek sayısının yetersiz olduğu durumlarda etkili olmayabilir.
Borderline-SMOTE	Veri dengesizliği sorununu çözmek için etkili bir yöntemdir. Bu yöntem sadece azınlık sınıfındaki örneklerle ilgilenir ve çoğunluk sınıfındaki örnekleri değiştirmez, bu da çoğunluk sınıfındaki örneklerin bilgilerinin kaybedilmesini önler. Borderline-SMOTE, SMOTE yöntemine göre daha az sentetik örnek üretir ve böylece daha az hata yapma olasılığı vardır. Bu yöntem, SMOTE yöntemi gibi, kullanıcının belirleyebileceği bir k değeri kullanarak sentetik örneklerin üretilmesini kontrol etme imkânı sağlar.	Borderline-SMOTE, bazı durumlarda yanlış sınıflandırılmış örneklerle sentetik örnekler üretebilir. Bu yöntem, azınlık sınıfındaki örneklerin konumuna bağlıdır. Azınlık sınıfındaki örnekler belirli bir alanı kaplamıyorsa, bu yöntem çok az veya hiç sentetik örnek üretemeyebilir. Borderline-SMOTE, azınlık sınıfındaki örneklerin orijinal dağılımını korumaz. Bu nedenle, bu yöntemle üretilen sentetik örnekler orijinal veri setine göre daha homojen olabilir. Borderline-SMOTE, bir veri setindeki örnek sayısı arttıkça daha fazla zaman alabilir. Bu nedenle, büyük veri setleri için uygun olmayabilir.
ADASYN	ADASYN, sınıf dengesizliği olan veri kümelerinde daha iyi bir sınıf dengesi sağlar. Bu nedenle, model performansını artırır ve daha doğru sonuçlar elde etmek için kullanılabilir. ADASYN, sentetik veri örnekleri üretirken gerçek veri örneklerinin dağılımını dikkate alır. Bu nedenle, sentetik veriler daha gerçekçi ve orijinal verilere daha yakındır. ADASYN, veri setindeki her sınıf için farklı sayıda yapay örnekler üretmek için bir ağırlıklandırma faktörü kullanır. Bu sayede, kullanıcılar yapay örneklerin sayısını kontrol edebilirler.	ADASYN, sentetik veri örnekleri üretirken orijinal verileri temel alır. Ancak, bu yapay örnekler, gerçek verilerle tamamen örtüşmeyebilir ve modelin yanlış öğrenmesine neden olabilir. ADASYN, daha küçük veri kümelerinde sınıf dengesizliği sorununu çözmek için uygun değildir. Bu nedenle, daha büyük veri kümelerinde kullanılması daha uygun olabilir. ADASYN, sentetik veri örnekleri üretmek için hesaplama yoğunluğu yüksek bir yöntemdir. Bu nedenle, büyük veri kümelerinde kullanıldığında hesaplama süresi artabilir.

Tablo 6'ın Devamı

Sentetik Veri Üretme Tekniği	Olumlu Yönleri	Olumsuz Yönleri
ROSE	ROSE, sınıf dengesizliği olan veri kümelerinde daha iyi bir sınıf dengesi sağlar. Bu nedenle, model performansını artırır ve daha doğru sonuçlar elde etmek için kullanılabilir. ROSE, uygulanması kolay ve basit bir yöntemdir. Bu nedenle, özellikle küçük ve orta boyutlu veri kümeleri için uygun bir seçenek olabilir. ROSE, kullanıcıların yapay örneklerin sayısını kontrol etmelerine olanak tanır. Bu sayede, özellikle aşırı örnekleme durumunda, yapay örneklerin sayısının kontrol edilerek modelin daha doğru sonuçlar üretmesi sağlanabilir.	ROSE, veri setindeki örnekleri kopyalayarak veya değiştirerek sentetik örnekler oluşturur. Bu nedenle, üretilen yapay örnekler gerçek verilere tam olarak benzemez ve yanıltıcı sonuçlara neden olabilir. ROSE, sentetik veri örnekleri üretirken orijinal veri setindeki örnekleri kopyalar veya değiştirir. Bu nedenle, veri boyutu önemli ölçüde artabilir ve veri setinin işlenmesi daha uzun sürebilir. ROSE, az örnekli sınıfların sentetik örnekleri üretirken fazla örnekleme yapabilir. Bu nedenle, az örnekli sınıfların veri dağılımı bozulabilir ve yanıltıcı sonuçlar elde edilebilir.
Trim-SMOTE	Trim-SMOTE, veri kümelerindeki sınıf dengesizliğini gidererek daha iyi bir sınıf dengesi sağlar. Bu da model performansını artırır ve daha doğru sonuçlar elde etmek için kullanılabilir. Trim-SMOTE, az örnekli sınıfların sayısını artırarak doğruluk ve hassasiyet gibi model performans ölçütlerini artırabilir. Trim-SMOTE, yapay örneklerin sayısı, minimum ve maksimum komşu sayısı gibi parametrelerin özelleştirilebilir olması sayesinde kullanıcıların ihtiyaçlarına göre uyarlanabilir.	Trim-SMOTE, sentetik örnekler oluşturarak veri setinin boyutunu artırır. Bu nedenle, veri kümesi büyüdüğünde, işlem süresi ve bellek gereksinimleri artabilir. Trim-SMOTE, az örnekli sınıfların verilerini kopyalayarak ve rastgele örnekler üreterek sentetik örnekler oluşturur. Bu nedenle, yanlış örneklerin üretilme riski vardır ve bu örnekler yanıltıcı sonuçlara neden olabilir. Trim-SMOTE, az örnekli sınıfların veri dağılımını bozarak sentetik örnekler üretebilir. Bu nedenle, az örnekli sınıfların özellikleri ve örneklerin dağılımı bozulabilir, bu da yanıltıcı sonuçlara neden olabilir.

Tablo 6'ın Devamı

Sentetik Veri Üretme Tekniği	Olumlu Yönleri	Olumsuz Yönleri
MWMOTE	MWMOTE, veri kümelerindeki sınıf dengesizliğini gidererek daha iyi bir sınıf dengesi sağlar. Bu da model performansını artırır ve daha doğru sonuçlar elde etmek için kullanılabilir. MWMOTE, az örnekli sınıfların özelliklerini koruyarak veri artırır. Bu sayede, az örnekli sınıfların özellikleri bozulmadan model performansı artırılabilir. MWMOTE, sentetik örnekler oluşturarak veri setinin boyutunu artırır. Bu sayede, modelin daha iyi öğrenmesine ve daha doğru sonuçlar üretmesine yardımcı olabilir.	MWMOTE, veri kümesindeki örneklerin sayısını artırarak işlem süresini ve eğitim zamanını artırır. Bu nedenle, büyük veri kümelerinde kullanılması durumunda eğitim zamanı uzun sürebilir. MWMOTE, az örnekli sınıfların verilerini kopyalayarak ve rastgele örnekler üreterek sentetik örnekler oluşturur. Bu nedenle, yanlış örneklerin üretilme riski vardır ve bu örnekler yanıltıcı sonuçlara neden olabilir. MWMOTE, az örnekli sınıfların özelliklerini koruyarak veri artırır. Ancak, bazı durumlarda az örnekli sınıfların özellikleri bozulabilir ve yanıltıcı sonuçlara neden olabilir. Bu nedenle, veri kümelerinin özellikleri dikkatli bir şekilde incelenmeli ve yöntemin uygulanacağı veri kümesine uygun parametreler seçilmelidir.
HAC	HAC, hiyerarşik bir kümeleme yöntemi olduğu için, kümeleme sonucunu ayrıntılı bir şekilde gösterir. Böylece, veri setindeki gruplar ve aralarındaki ilişkiler hakkında daha iyi bir anlayış sağlar. HAC, veri setindeki doğal yapıları ortaya çıkarmak için kullanılabilir. Bu sayede, veri setindeki anlamlı yapılar belirlenerek, daha etkili bir veri analizi ve model oluşturma yapılabilir. HAC, küme sayısını belirlemek için kullanılabilir. Bu sayede, veri setindeki anlamlı gruplara bölme yaparak, daha iyi bir model performansı elde edilebilir.	HAC, tüm veri örneklerini bir arada tuttuğu için, büyük veri setleri üzerinde hesaplama gücü açısından sınırlıdır. Bu nedenle, büyük veri setleri üzerinde kullanılırken, işlem zamanı uzun olabilir. HAC, küme merkezleri arasındaki mesafeyi ölçerek gruplar oluşturur. Bu mesafeler, veri setindeki veri örnekleri üzerindeki hatalar ve veri kayıpları nedeniyle değişebilir. Bu nedenle, HAC duyarlı bir yöntemdir ve farklı veri örnekleri üzerinde farklı sonuçlar üretebilir. HAC, veri setindeki aykırı veriler (outlier) nedeniyle sonuçları etkilenebilir. Aykırı veriler, gruplamaları ve kümeleme sonuçlarını olumsuz yönde etkileyebilir. Bu nedenle, HAC kullanmadan önce, aykırı verilerin tespiti ve eleme işlemleri yapılmalıdır.

Tablo 6'ın Devamı

Sentetik Veri Üretim Tekniği	Olumlu Yönleri	Olumsuz Yönleri
MDO	MDO, birden fazla tasarım alanını optimize etmek için kullanılabilir. Bu sayede, tasarım alanları arasındaki etkileşimler dikkate alınarak, daha iyi tasarım sonuçları elde edilebilir. MDO, kaynakların optimize edilmesi için kullanılabilir. Bu sayede, tasarım sürecindeki kaynak kullanımı azaltılabilir ve maliyetler düşürülebilir. MDO, tasarım sürecini optimize ederek, daha kısa bir tasarım süresi ve daha verimli bir tasarım sonucu elde edilebilir.	MDO, birden fazla tasarım alanını dikkate alarak, çok karmaşık bir optimizasyon süreci gerektirir. Bu nedenle, MDO kullanımı, tasarım sürecinde karmaşıklığı artırabilir. MDO, birden fazla tasarım alanını dikkate aldığı için, büyük veri hacimleriyle çalışır. Bu nedenle, veri toplama ve işleme süreci zaman alıcı ve zorlu olabilir. MDO, çok disiplinli takım çalışmasını gerektirir. Bu nedenle, tasarım ekibi arasında iyi bir iş birliği ve koordinasyon gereklidir. Aksi takdirde, MDO yönteminin etkinliği azalabilir.

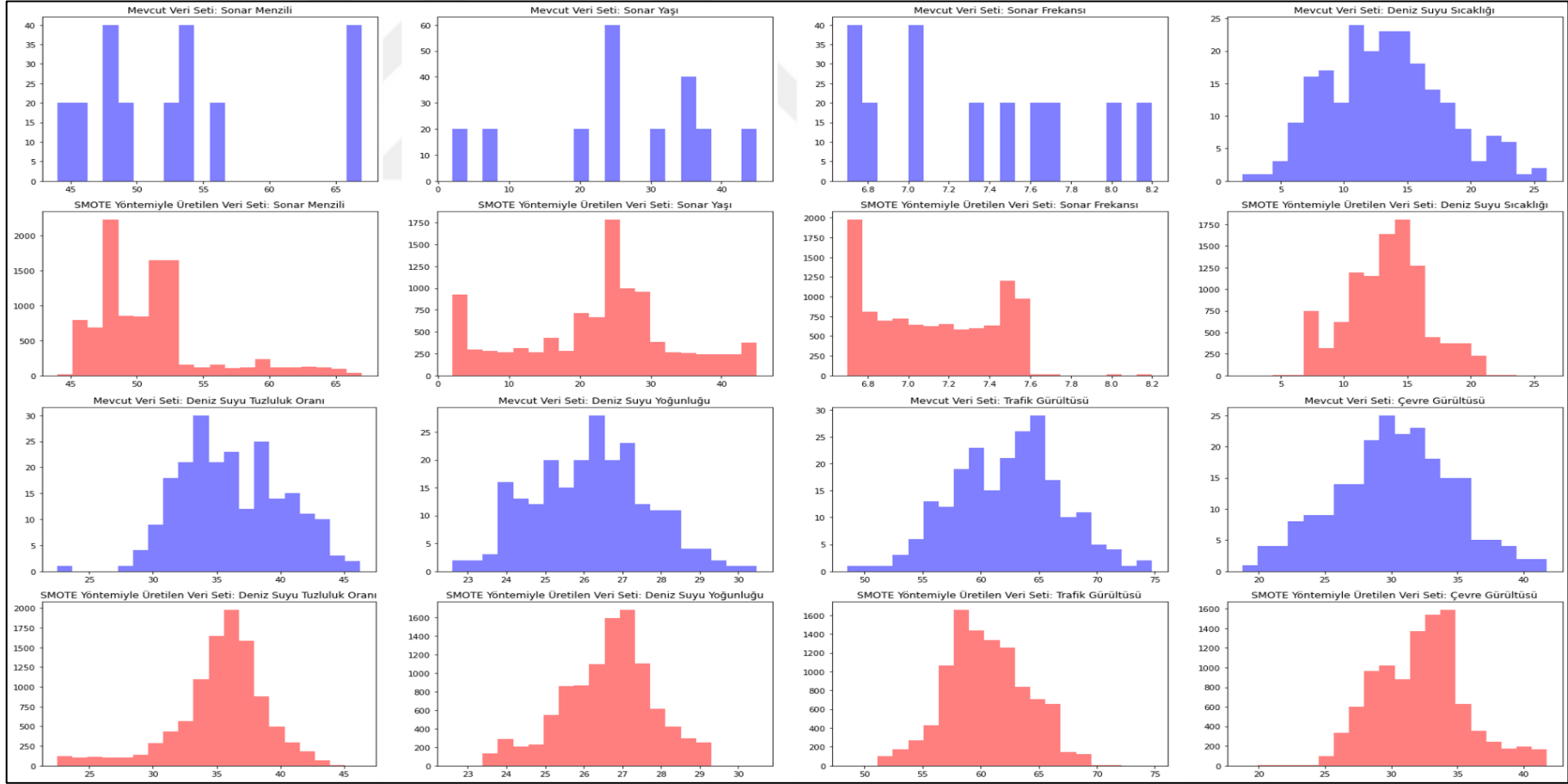
2.2.2 Sentetik Veri Üretme Tekniklerinin Karşılaştırılması

Yukarıda açıklanan her bir yöntem ile veri sayısı 10000 olan yeni veri setleri oluşturulmuştur. Ardından en iyi sentetik veri üretme tekniğinin belirlenmesi için detayları müteakip maddelerde açıklanan bir dizi testler uygulanmıştır.

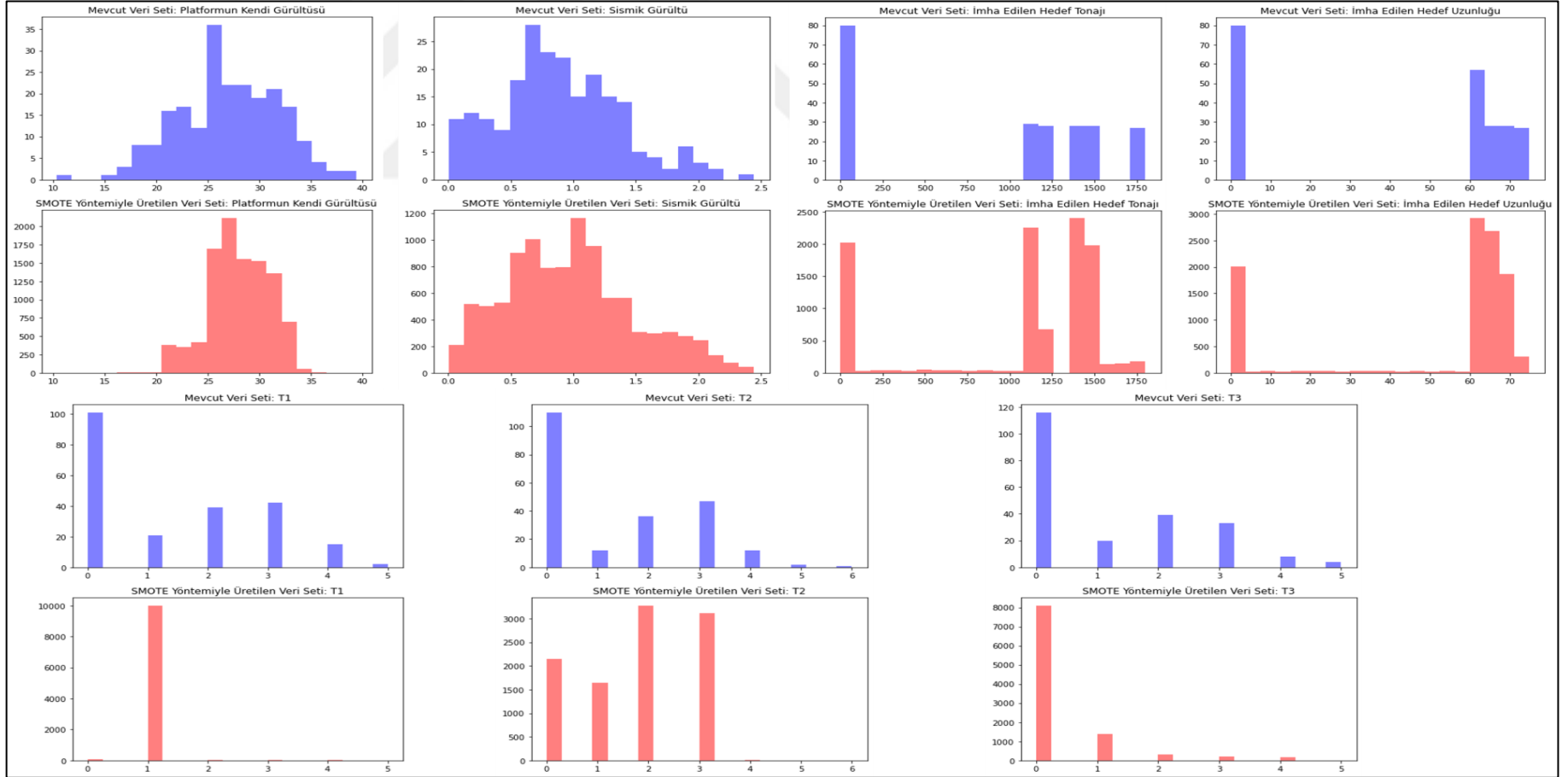
2.2.2.1 Histogram ve Yoğunluk Grafiklerinin İncelenmesi

Histogram yoğunluk grafikleri, veri setlerinin dağılımını görselleştirmek için kullanılan bir grafik türüdür. Bu grafikler, bir değişkenin değerlerinin frekansını gösterir ve bu frekansları belirli aralıklara bölerek her aralıktaki veri noktalarının sayısını ifade eder. Histogramlar, özellikle sayısal verilerin dağılımını anlamak ve analiz etmek için yaygın olarak kullanılır. Histogramlar, veri setindeki anormallikleri ve eğilimleri belirlemek için de kullanılabilir. Aykırı değerleri tespit etmek, genel eğilimleri değerlendirmek ve veri setindeki desenleri incelemek için histogramlar önemli bir araçtır. Ayrıca, karar verme süreçlerinde bilgi sağlamak ve veri setinin genel yapısını anlamak için kullanılırlar. Histogramlar aynı zamanda istatistiksel analizlerde ve araştırmalarda kullanılan temel bir görselleştirme aracıdır. Veri setinin genel dağılımını hızlı bir şekilde anlamak, örüntüler ve ilişkiler hakkında ilk gözlemleri yapmak için histogramlar etkili bir şekilde kullanılabilir. Histogram yoğunluk grafikleri, veri setlerinin dağılımını görsel olarak temsil etmek ve analiz etmek için kullanılan önemli bir araçtır. İstatistiksel analizlerde, araştırmalarda ve karar verme süreçlerinde bu grafik türü, verilerin genel özelliklerini anlamak ve içerdikleri bilgileri hızlı bir şekilde değerlendirmek için yaygın olarak kullanılmaktadır [43, 44].

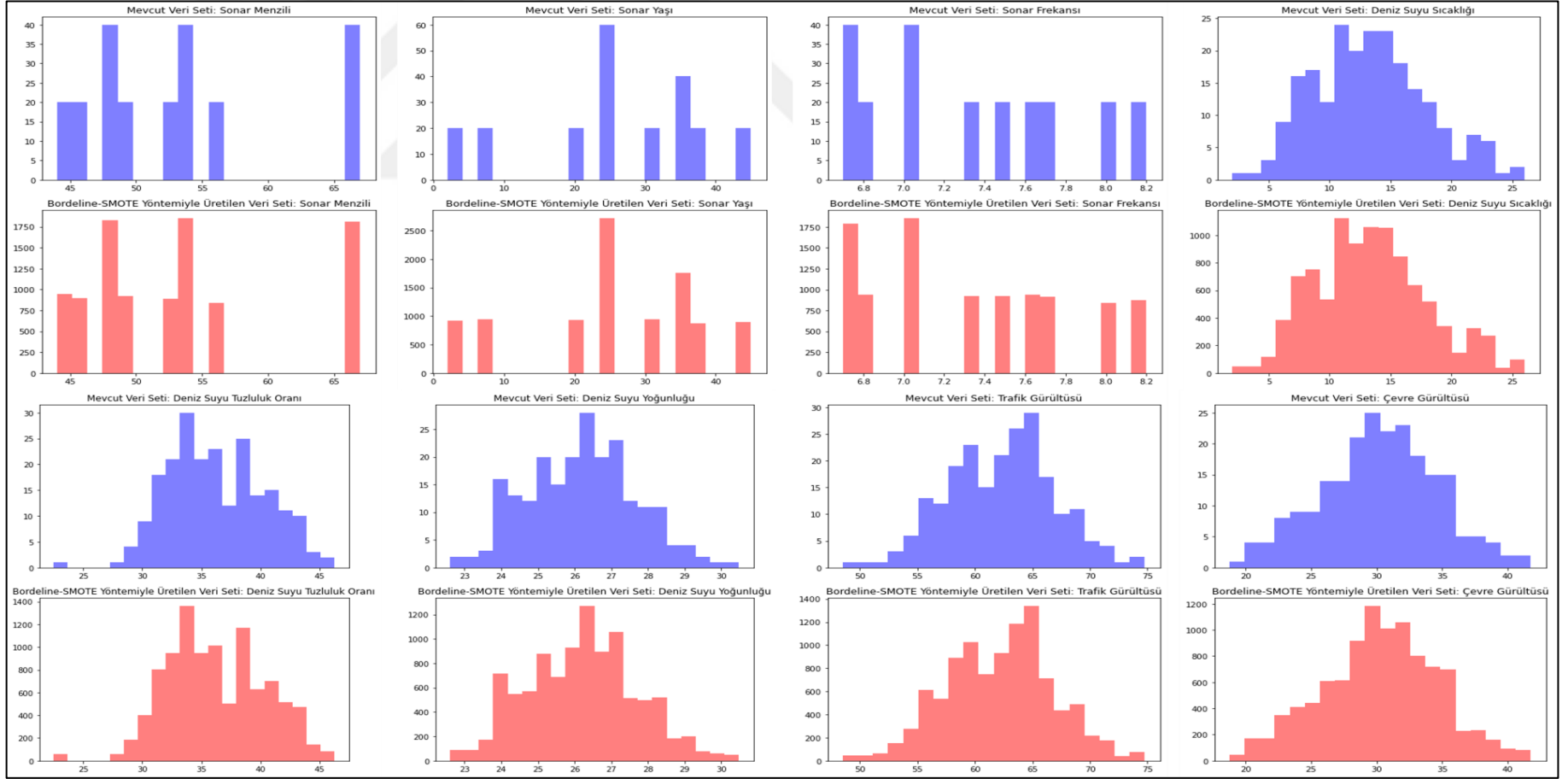
Bu kapsamda oluşturulan yeni veri setlerinin histogram ve yoğunluk grafikleri çizdirilerek her bir veri seti teker teker incelenmiştir. Bahse konu grafikler Şekil (4-19)'da sunulmuştur.



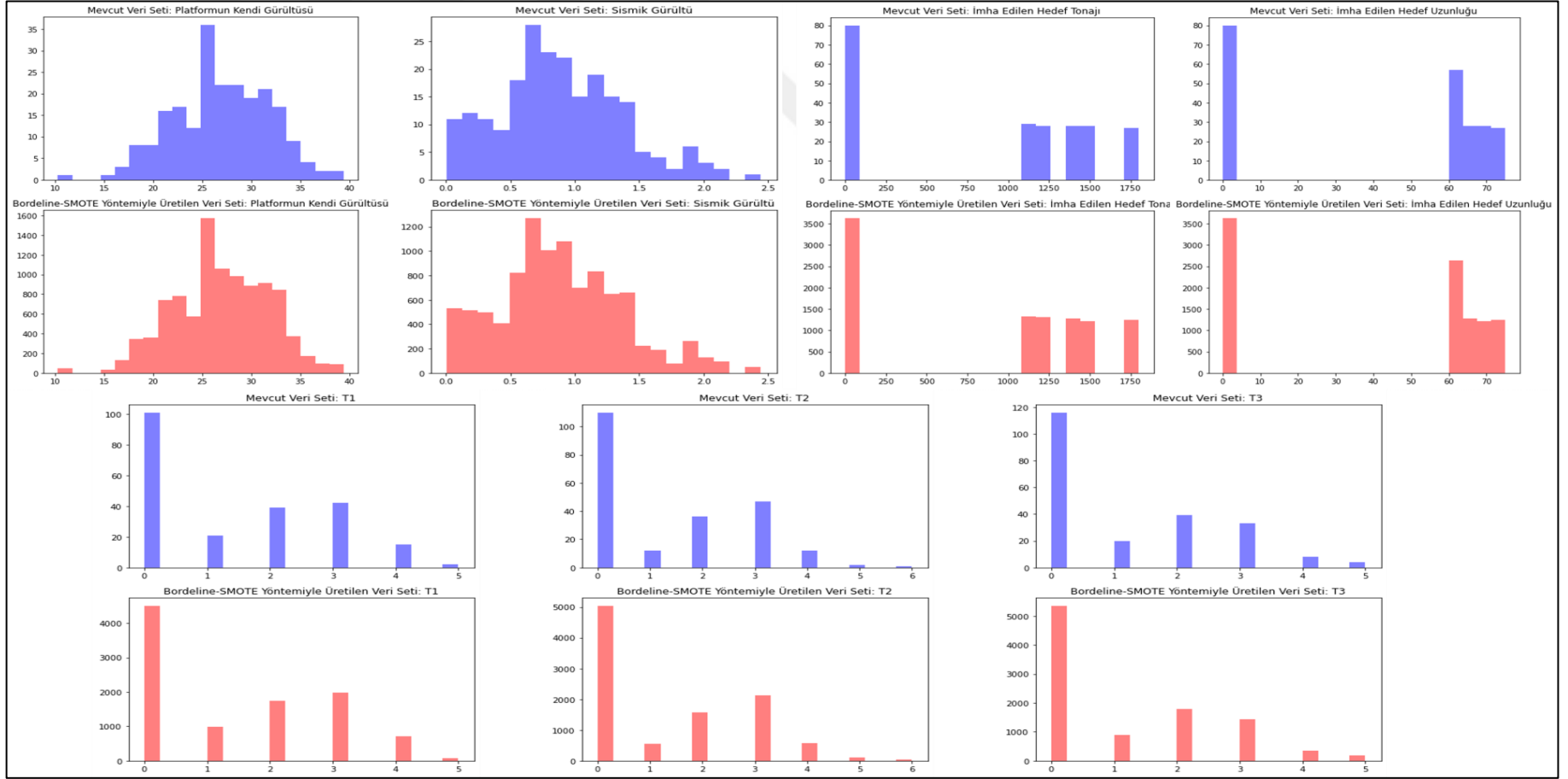
Şekil 4: SMOTE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-1



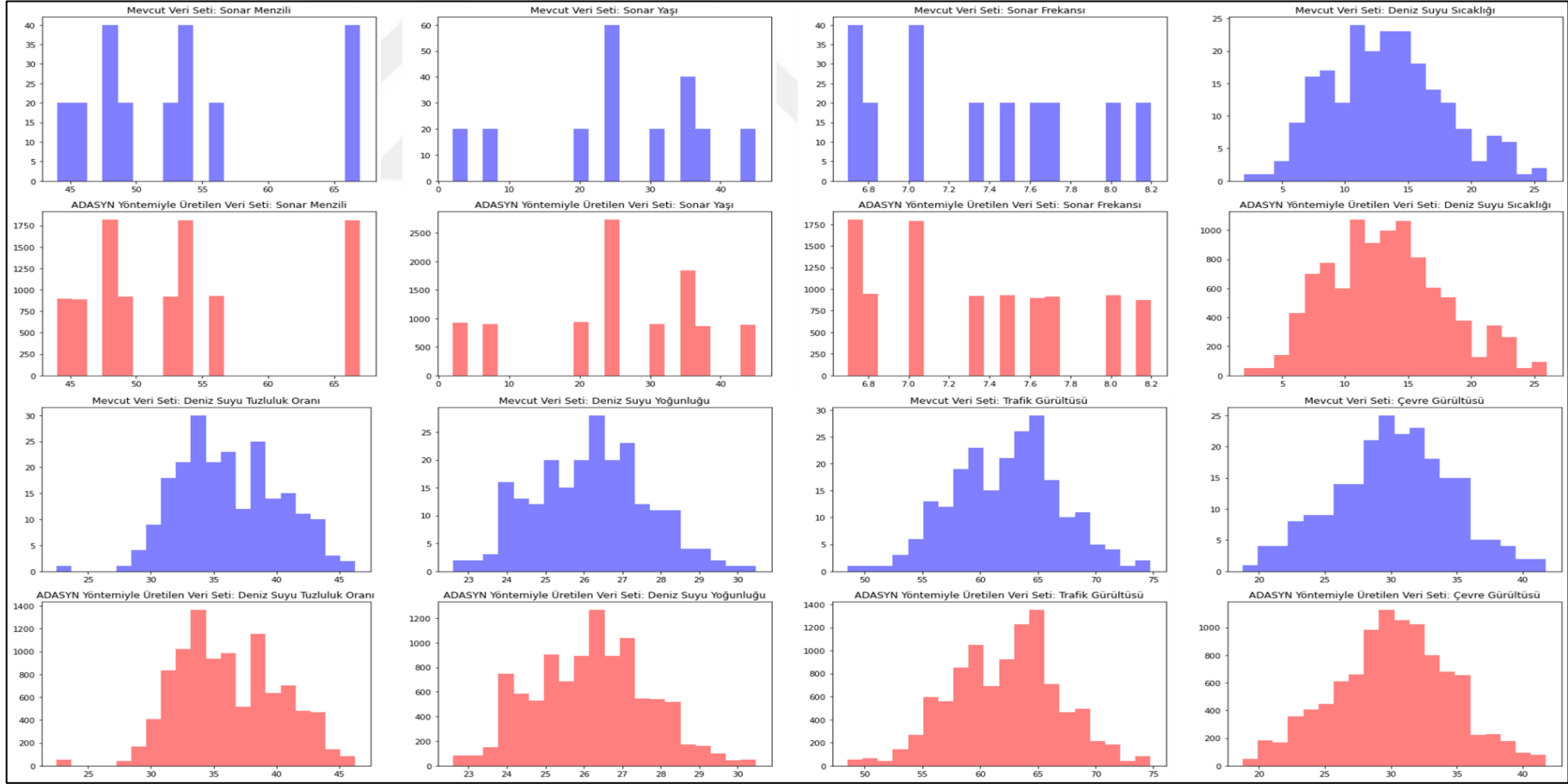
Şekil 5: SMOTE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-2



Şekil 6: Borderline SMOTE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-1



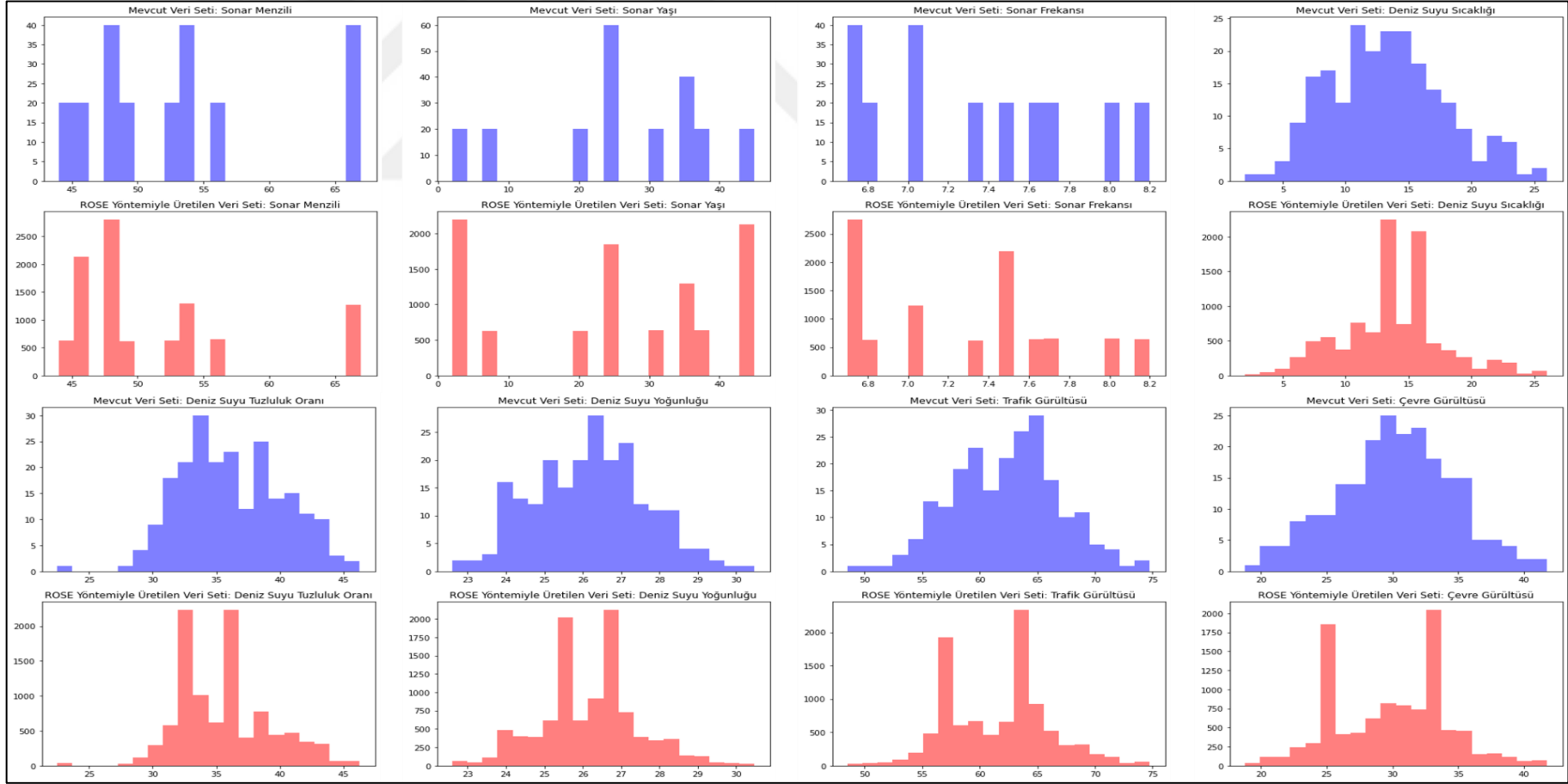
Şekil 7: Borderline SMOTE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-2



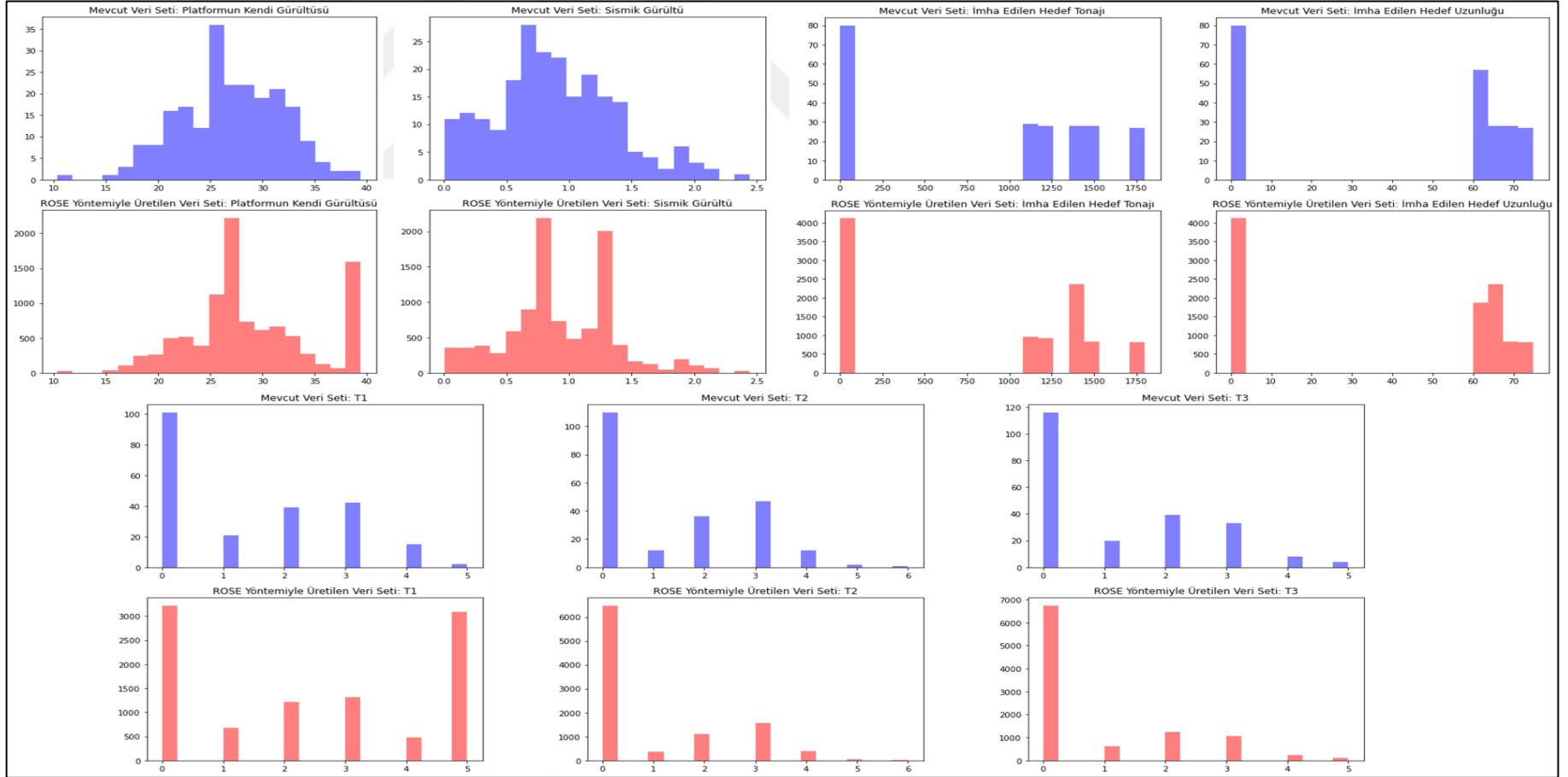
Şekil 8: ADASYN Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-1.



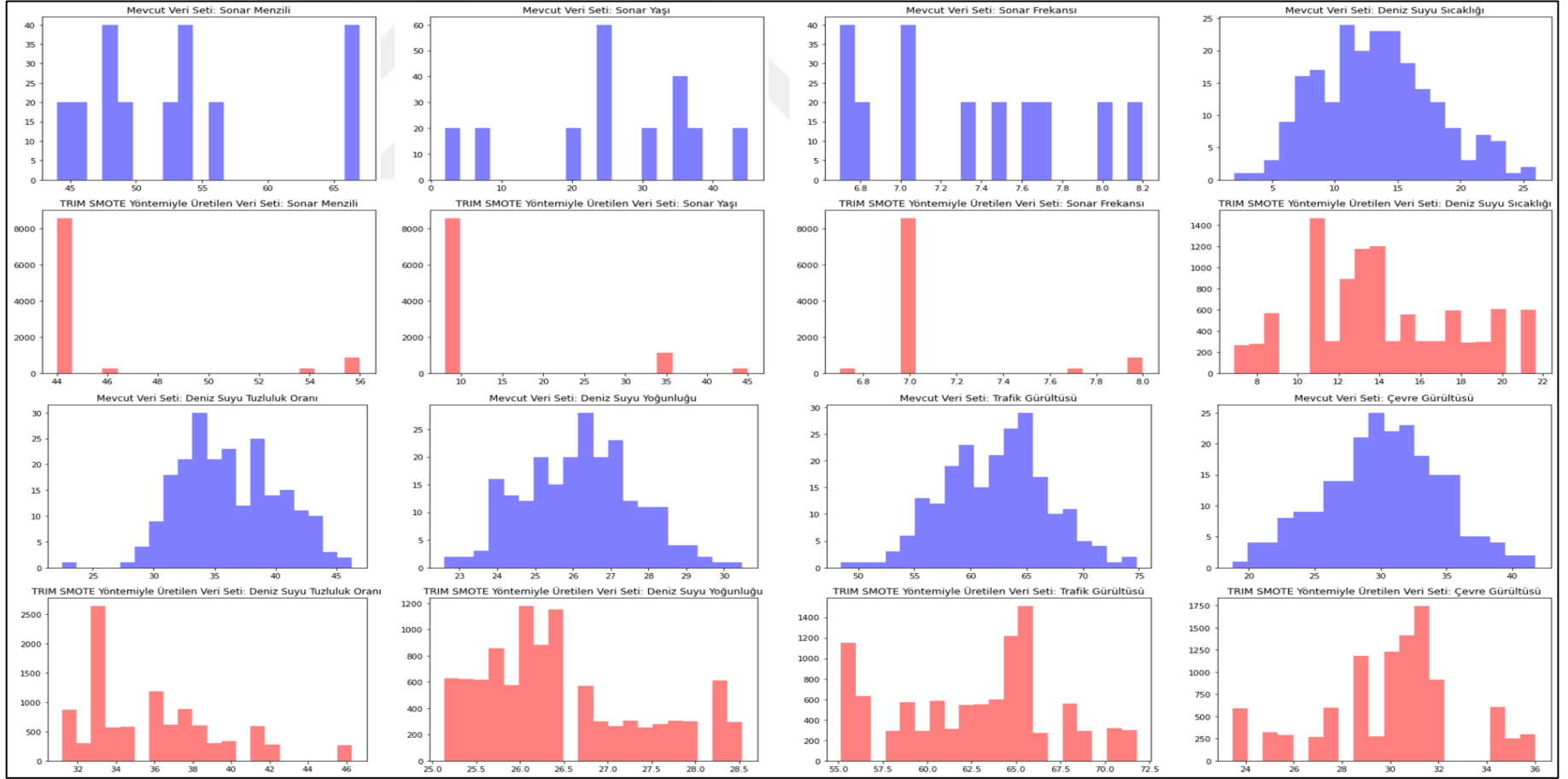
Şekil 9: ADASYN Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-2.



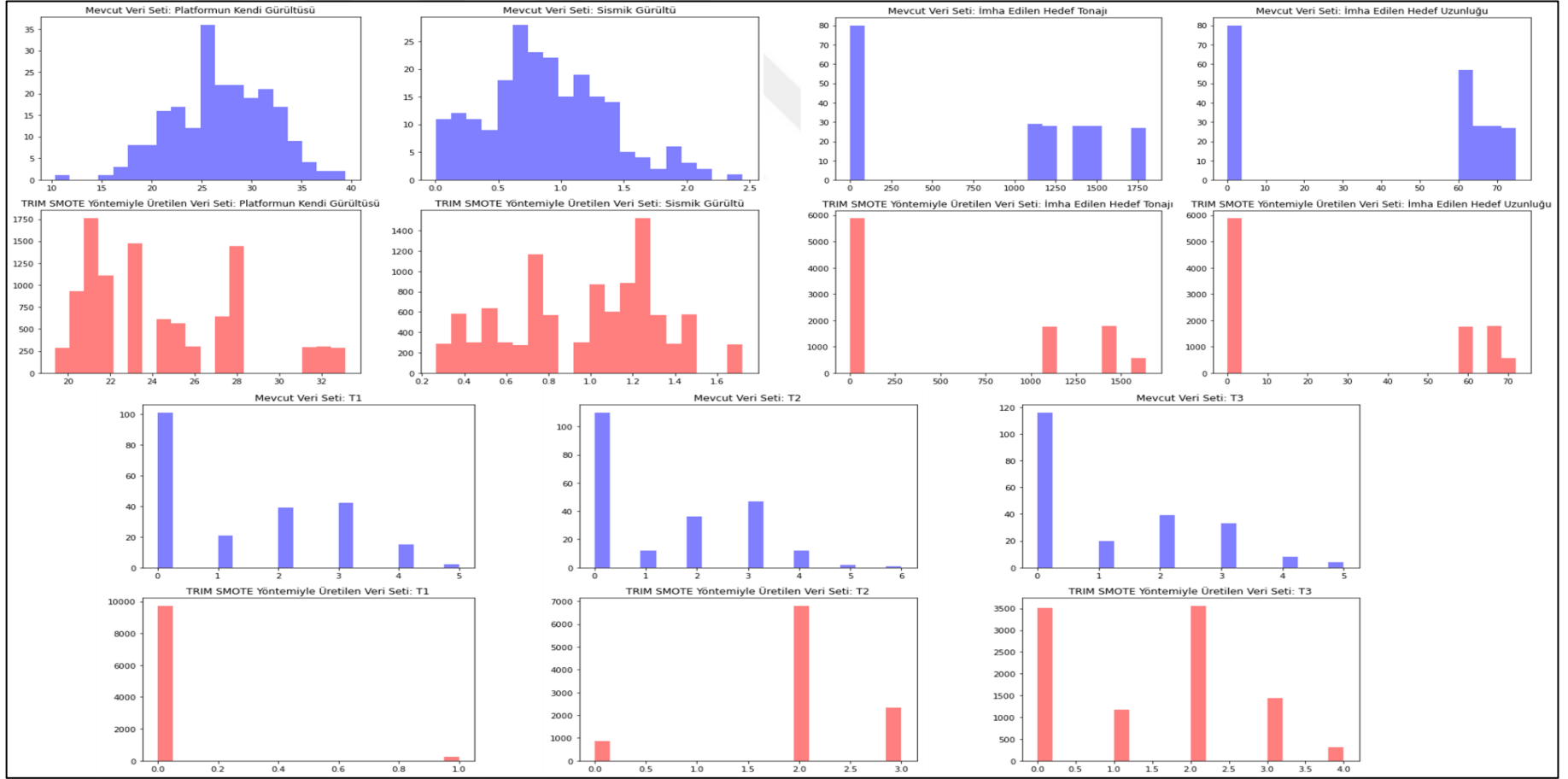
Şekil 10: ROSE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-1.



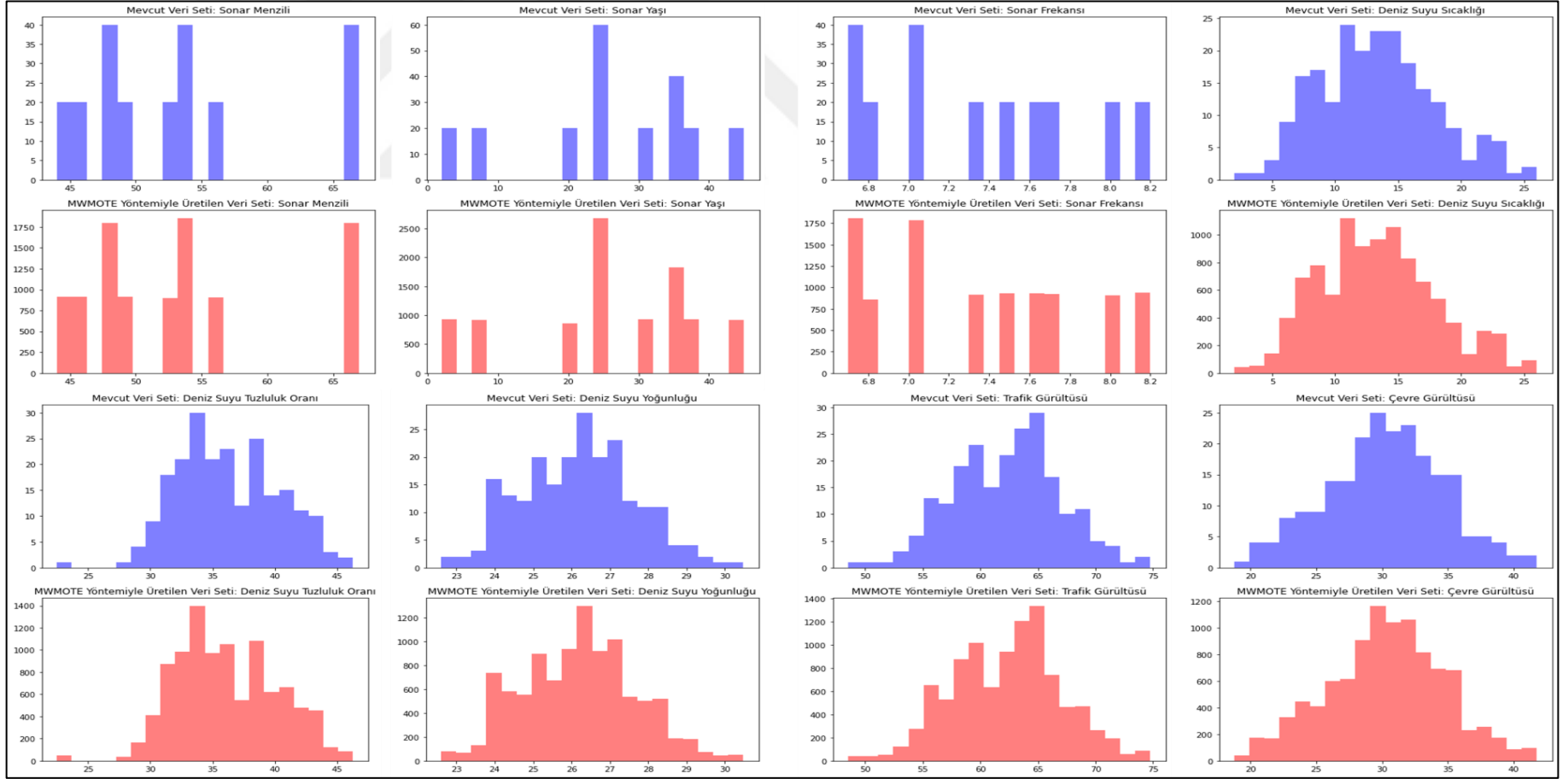
Şekil 11: ROSE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-2.



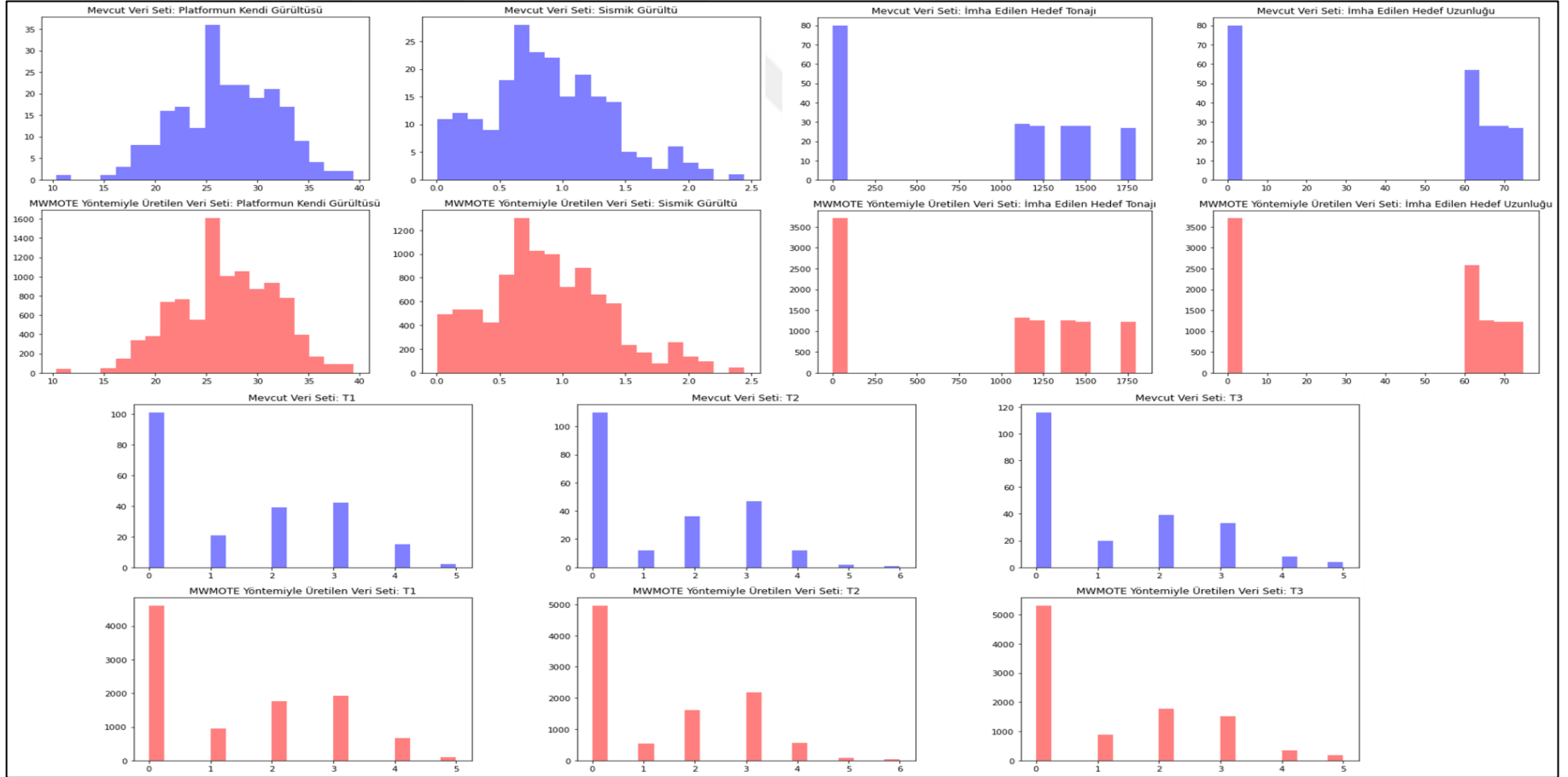
Şekil 12: Trim-SMOTE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-1.



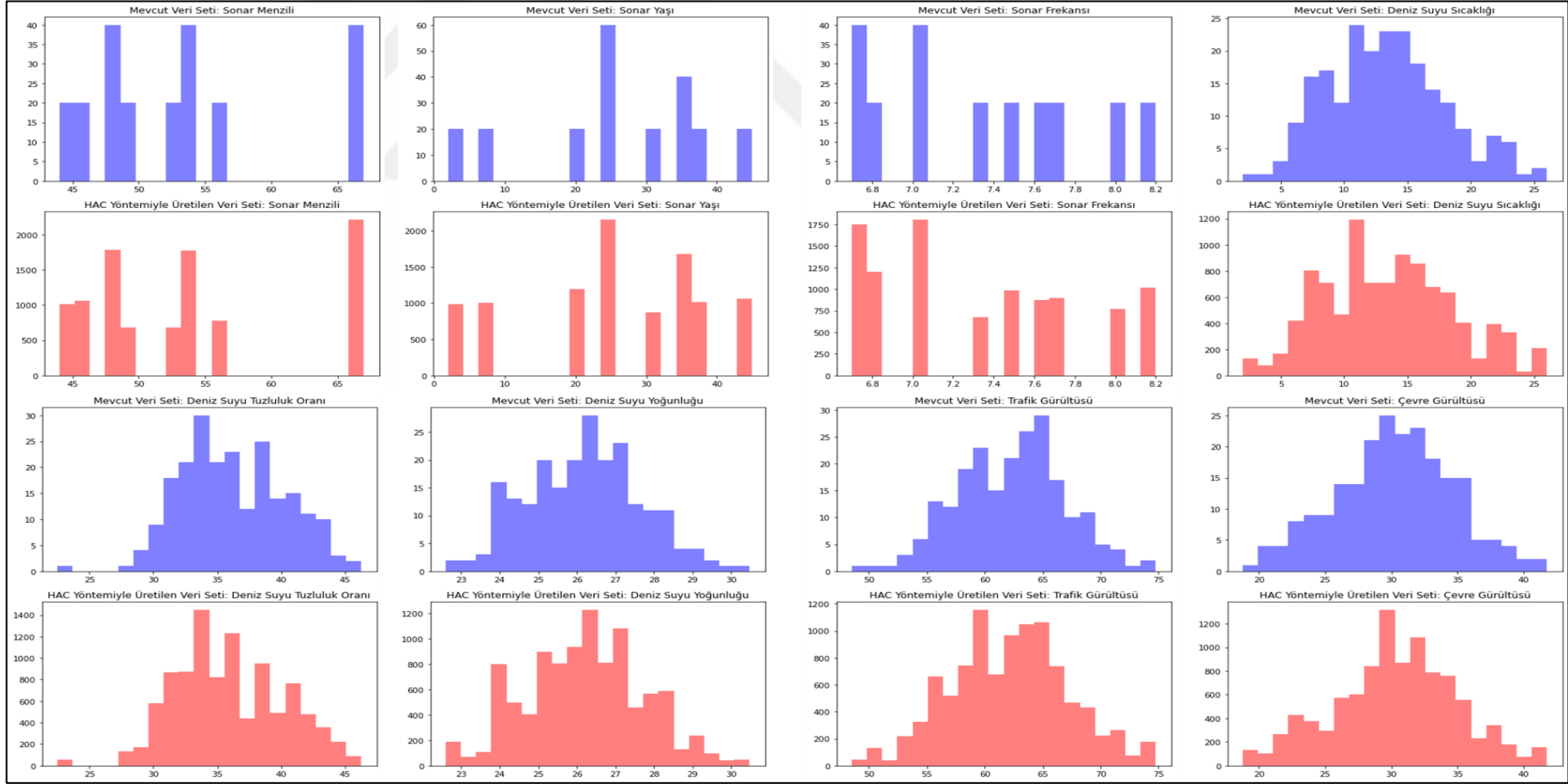
Şekil 13: Trim-SMOTE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-2.



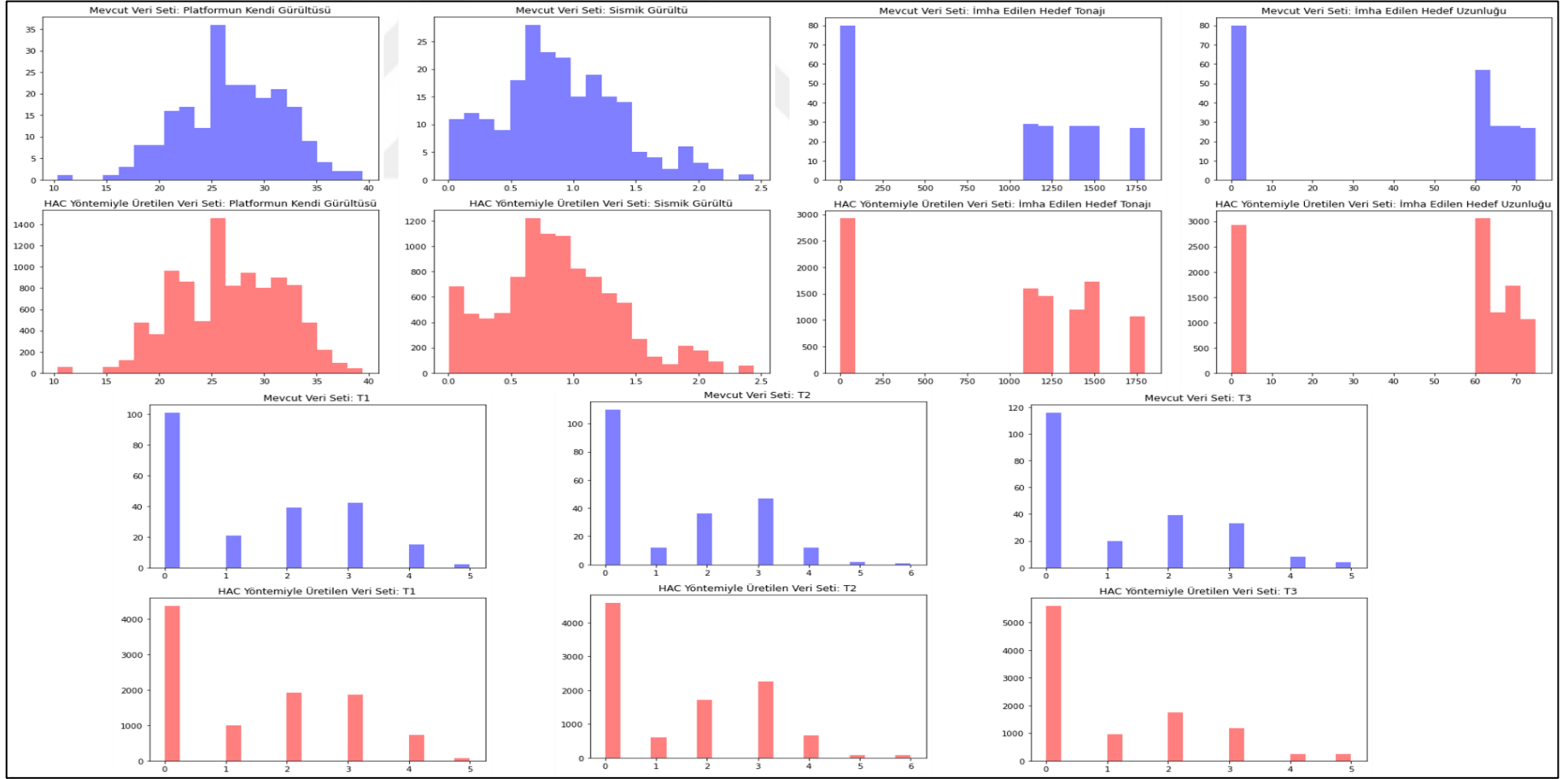
Şekil 14: MWMOTE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-1.



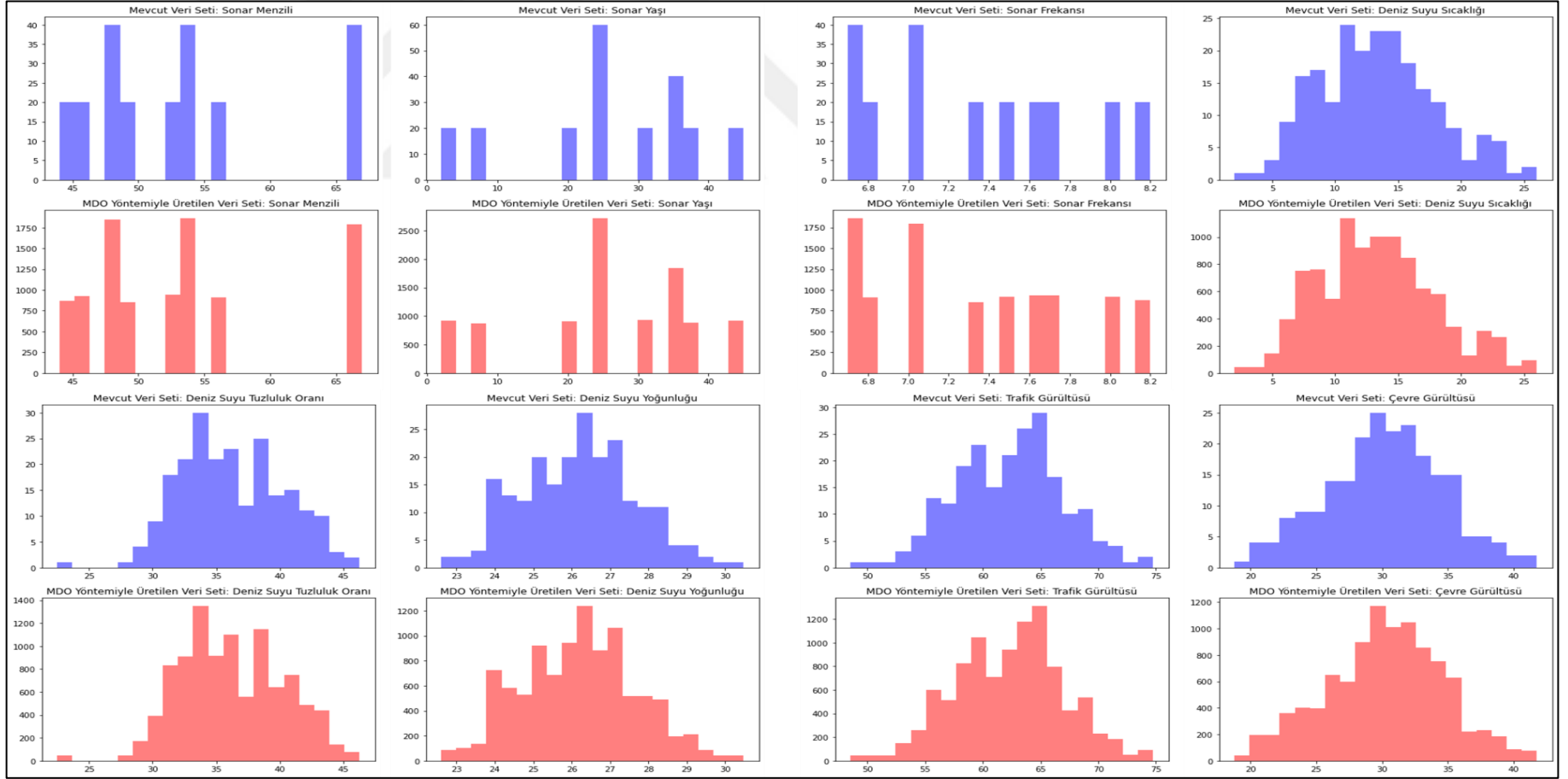
Şekil 15: MWMOTE Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-2.



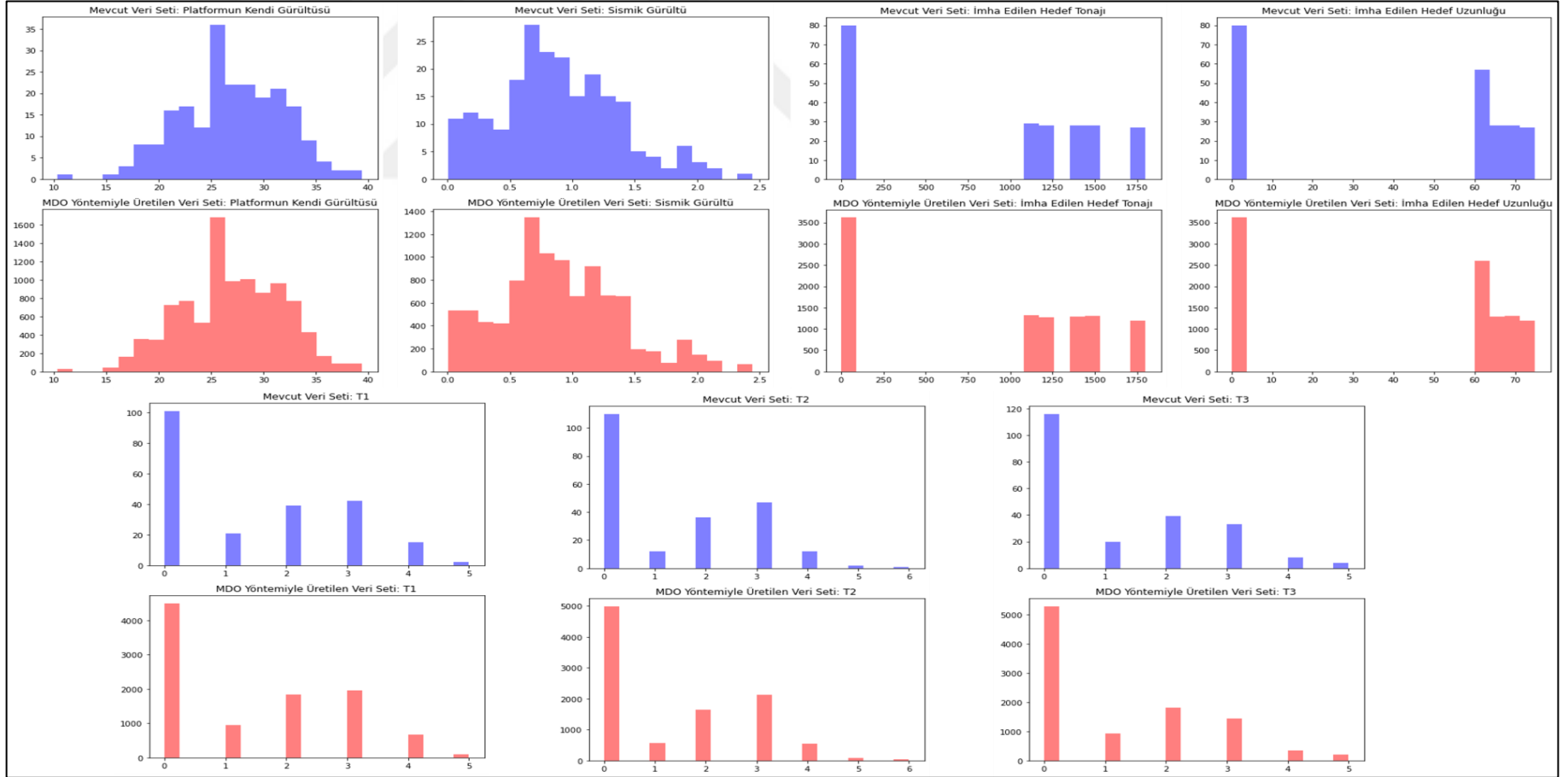
Şekil 16: HAC Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-1.



Şekil 17: HAC Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-2.



Şekil 18: MDO Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-1.



Şekil 19: MDO Yöntemiyle Üretilen Verilerin Histogram ve Yoğunluk Grafikleri-2.

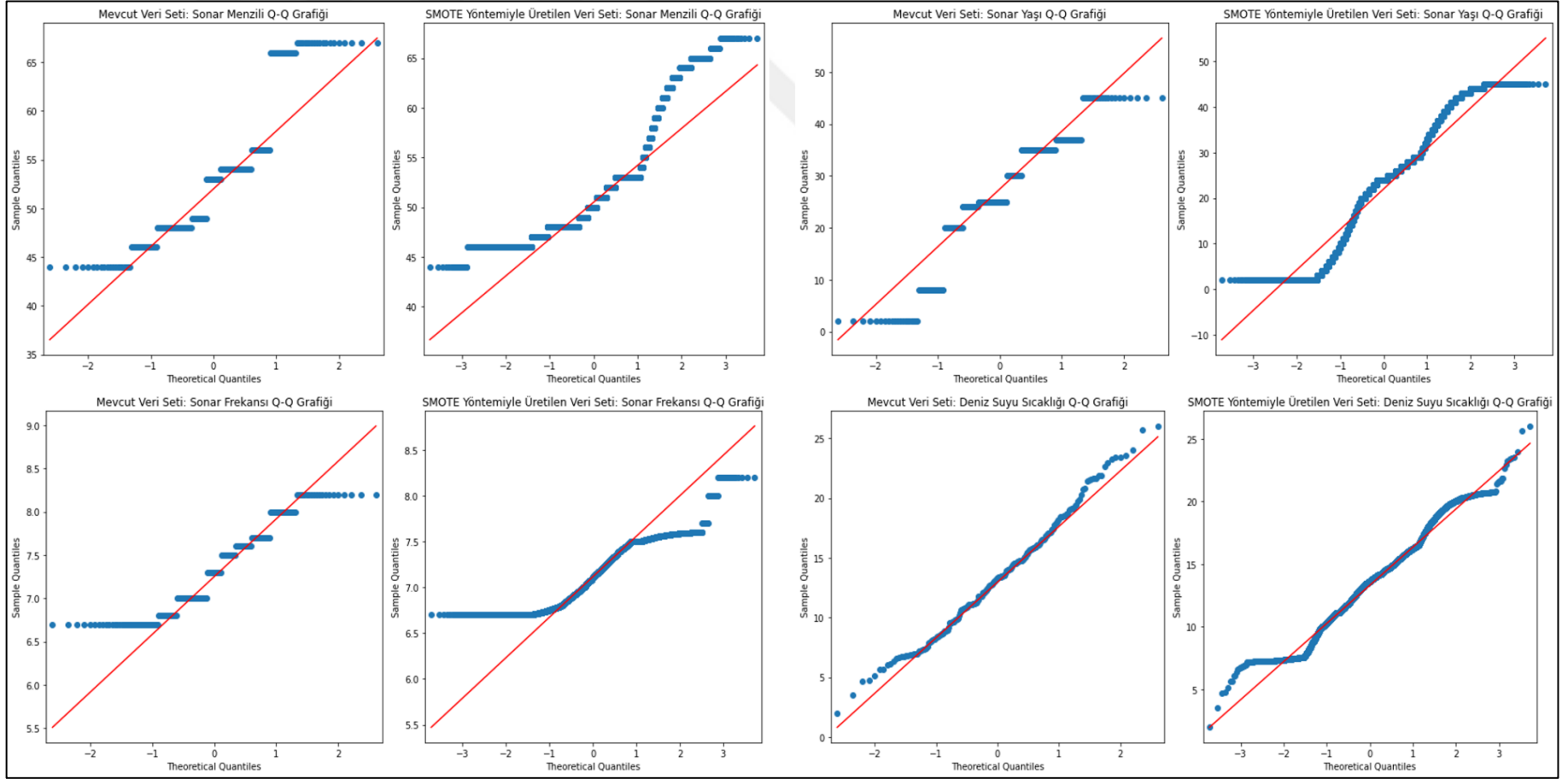
Şekil (4-19)'da yer alan grafikler incelendiğinde;

- Borderline SMOTE, ADASYN, MWMOTE, HAC ve MDO yöntemleriyle üretilen sentetik verilerin dağılımların kaynak veri setiyle oldukça benzer yapılara sahip oldukları,
- SMOTE, ROSE ve Trim-SMOTE yöntemleriyle üretilen sentetik verilerin dağılımların ise kaynak veri setiyle oldukça farklı yapılara sahip oldukları görülmektedir.

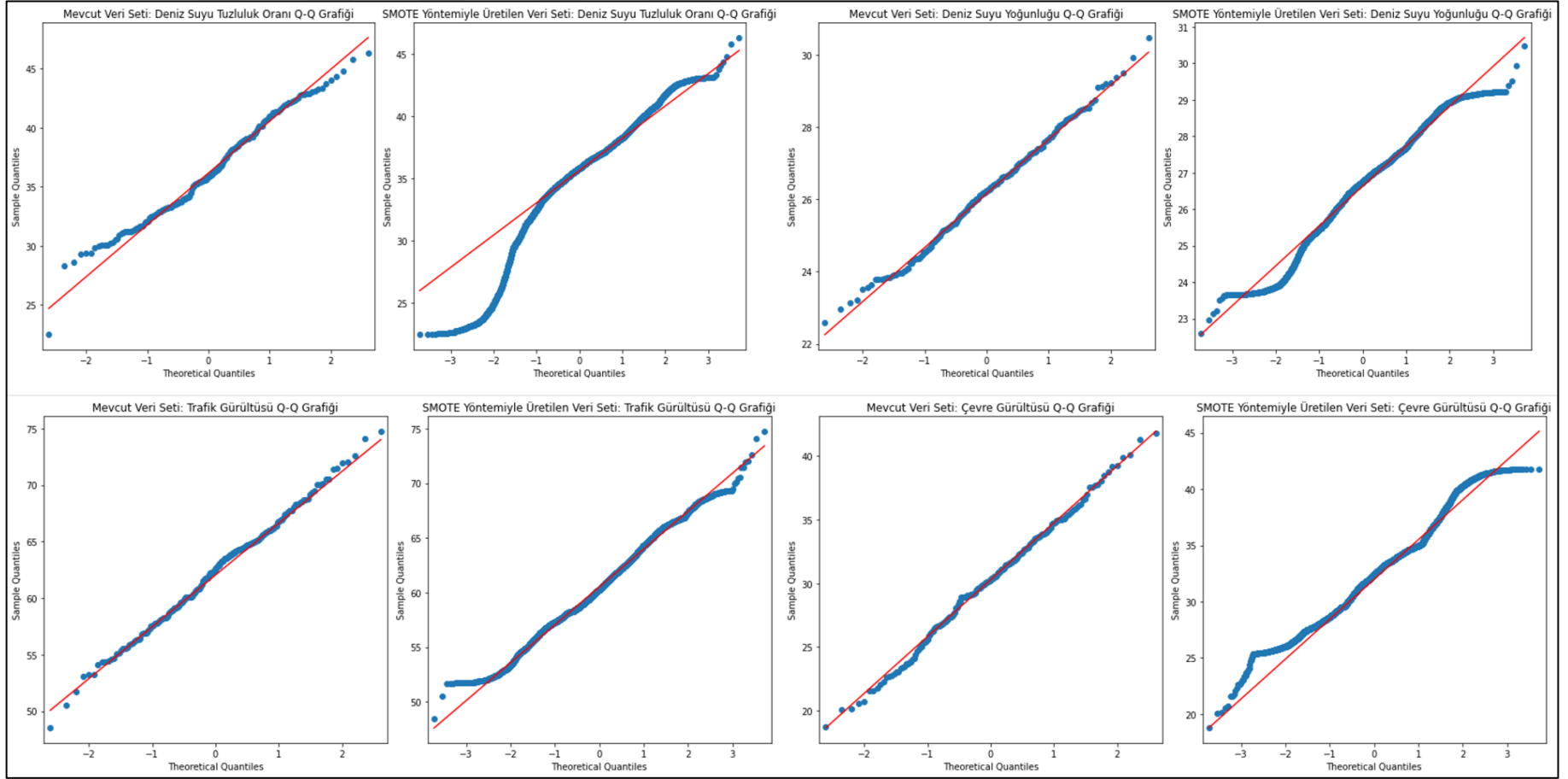
2.2.2.2 Q-Q Çizgilerinin İncelenmesi

Q-Q çizgileri, istatistiksel grafiklerde kullanılan ve iki değişken arasındaki ilişkiyi görselleştirmek için kullanılan grafiksel bir yöntemdir. "Q-Q" terimi, "quantile-quantile"nin kısaltmasıdır. Bu çizgiler, bir teorik dağılım ile gözlemlenen veriler arasındaki uyumu değerlendirmek için kullanılır. Q-Q çizgileri, normal dağılım gibi belirli bir teorik dağılıma sahip olan bir değişkenin, gerçek verilerle ne kadar iyi eşleştiğini gösterir. Çizgi, teorik ve gözlemlenen değerlerin sıralanmış quantile'larını karşılaştırarak çizilir. Eğer çizgi, 45 derecelik bir doğru üzerinde ise, bu demektir ki gözlemlenen veriler teorik dağılıma mükemmel bir şekilde uymaktadır. Q-Q çizgileri, özellikle istatistiksel analizlerde normal dağılım varsayımının geçerli olup olmadığını kontrol etmek için sıkça kullanılır. Normal dağılım, birçok istatistiksel testin temel varsayımıdır, bu nedenle Q-Q çizgileri, verilerin normal dağılıma ne kadar yakın olduğunu değerlendirmek için önemlidir. Ayrıca, Q-Q çizgileri, farklı dağılımları karşılaştırmak ve veri dönüşümleri üzerinde çalışmak için de kullanılabilir. Bu çizgiler aynı zamanda aykırı değerleri tespit etmek veya veri setindeki diğer önemli özellikleri görselleştirmek için de kullanılabilir. Q-Q çizgileri, istatistiksel analizlerin güvenilirliğini ve doğruluğunu artırmak için önemli bir araçtır [45, 46].

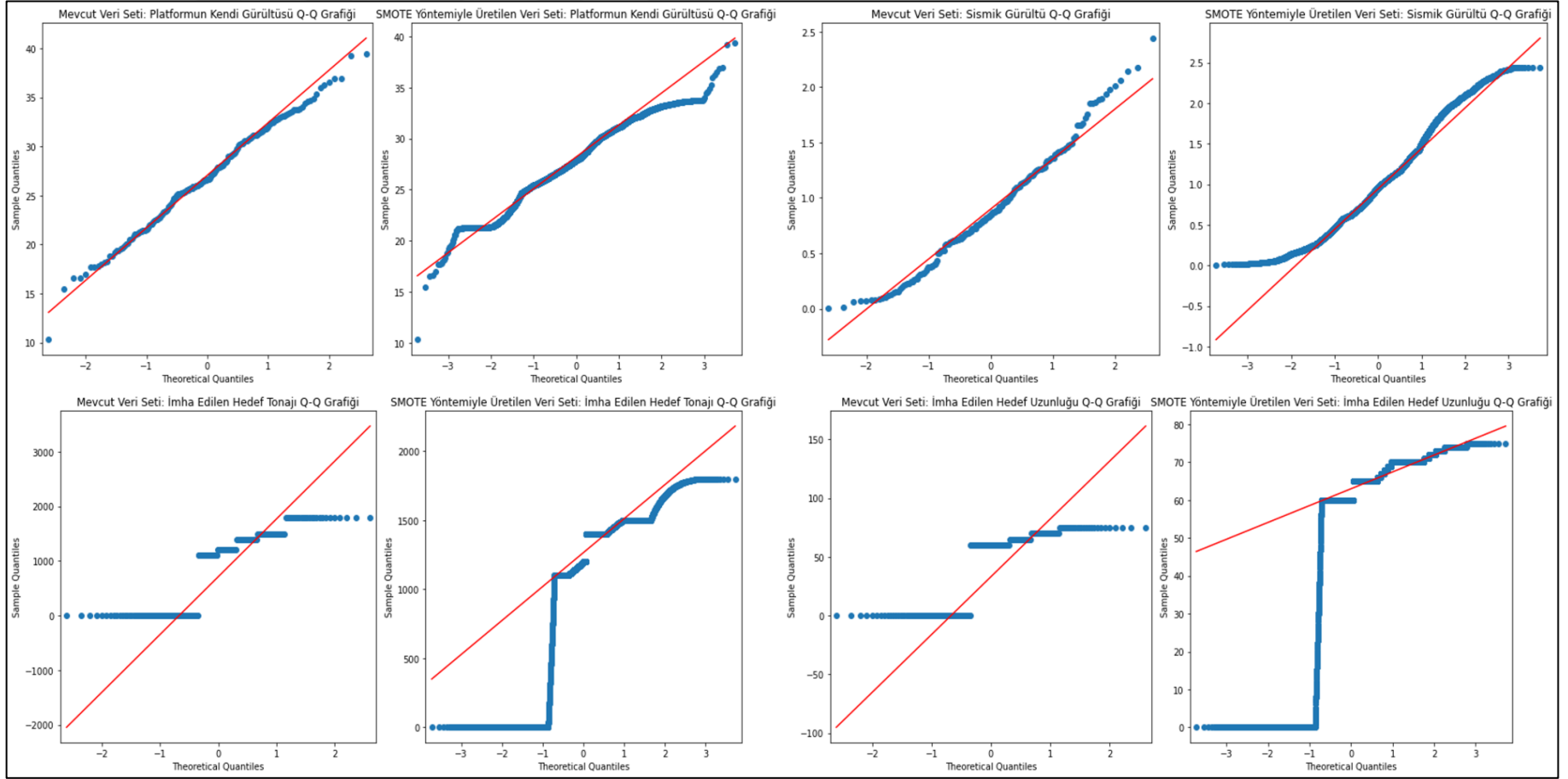
Bu kapsamda oluşturulan yeni veri setlerinin her birinin Q-Q çizgileri teker teker incelenmiştir. Bahse konu grafikler Şekil (20-51)'de sunulmuştur.



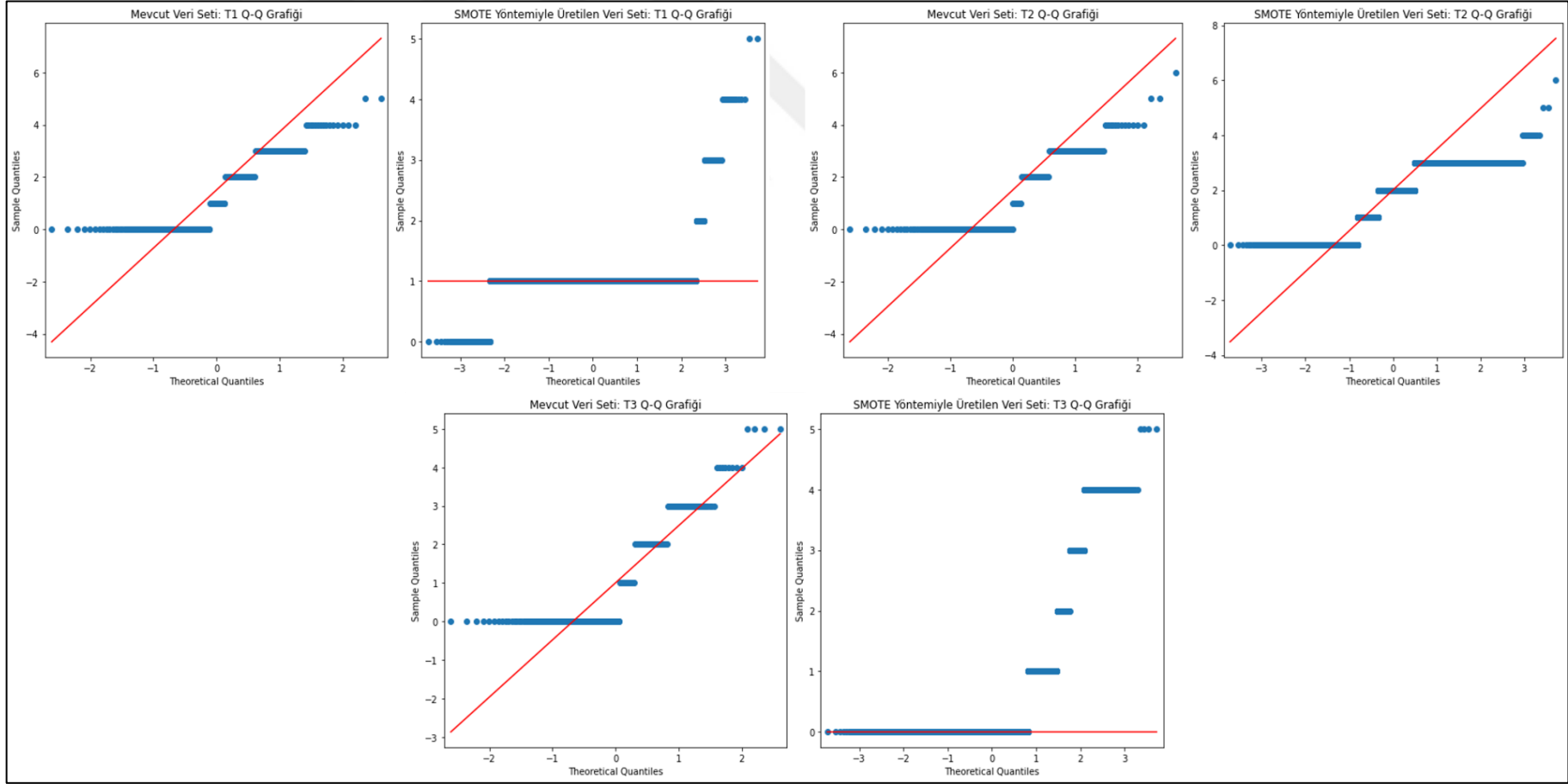
Şekil 20: SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-1.



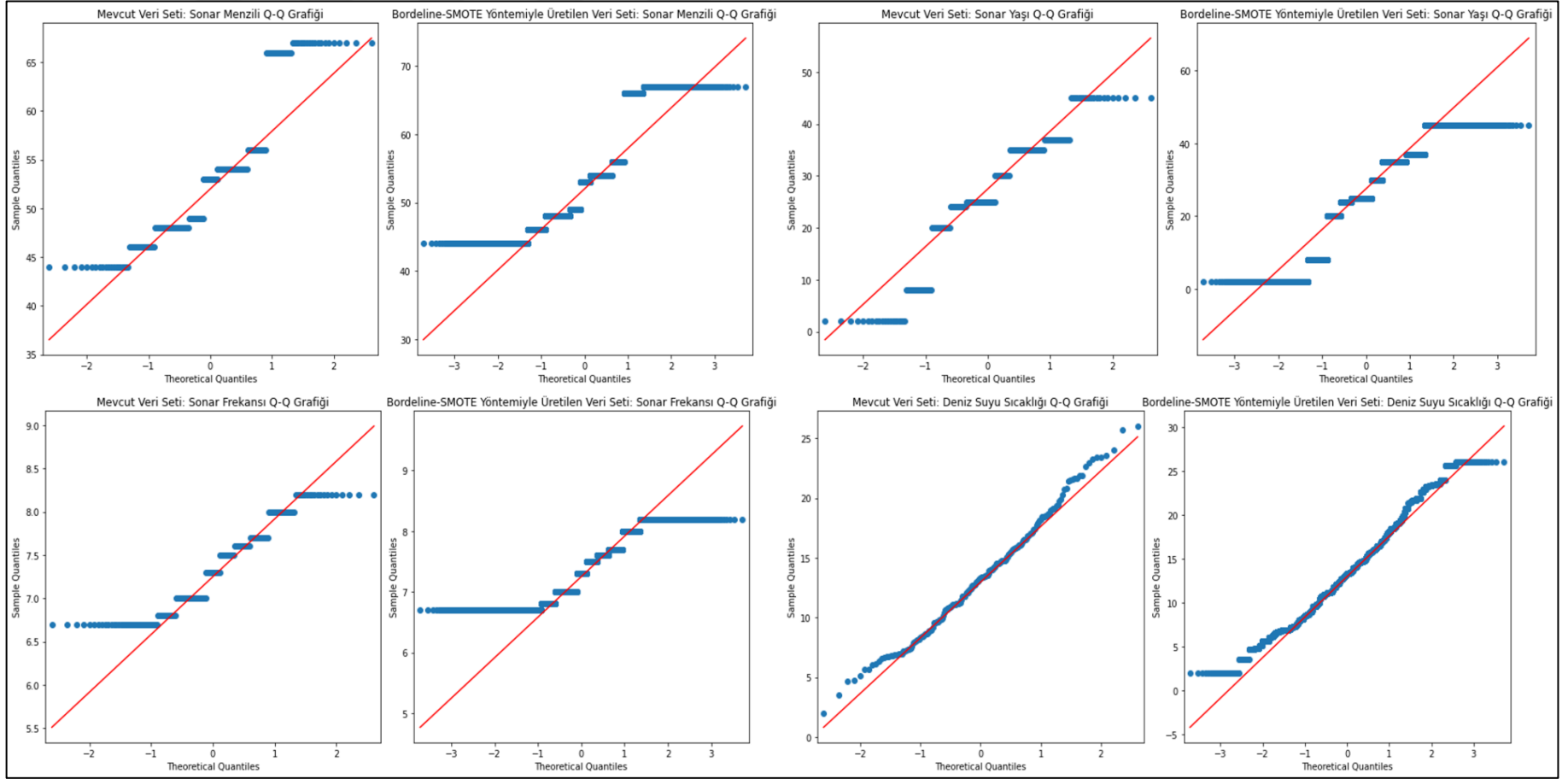
Şekil 21: SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-2.



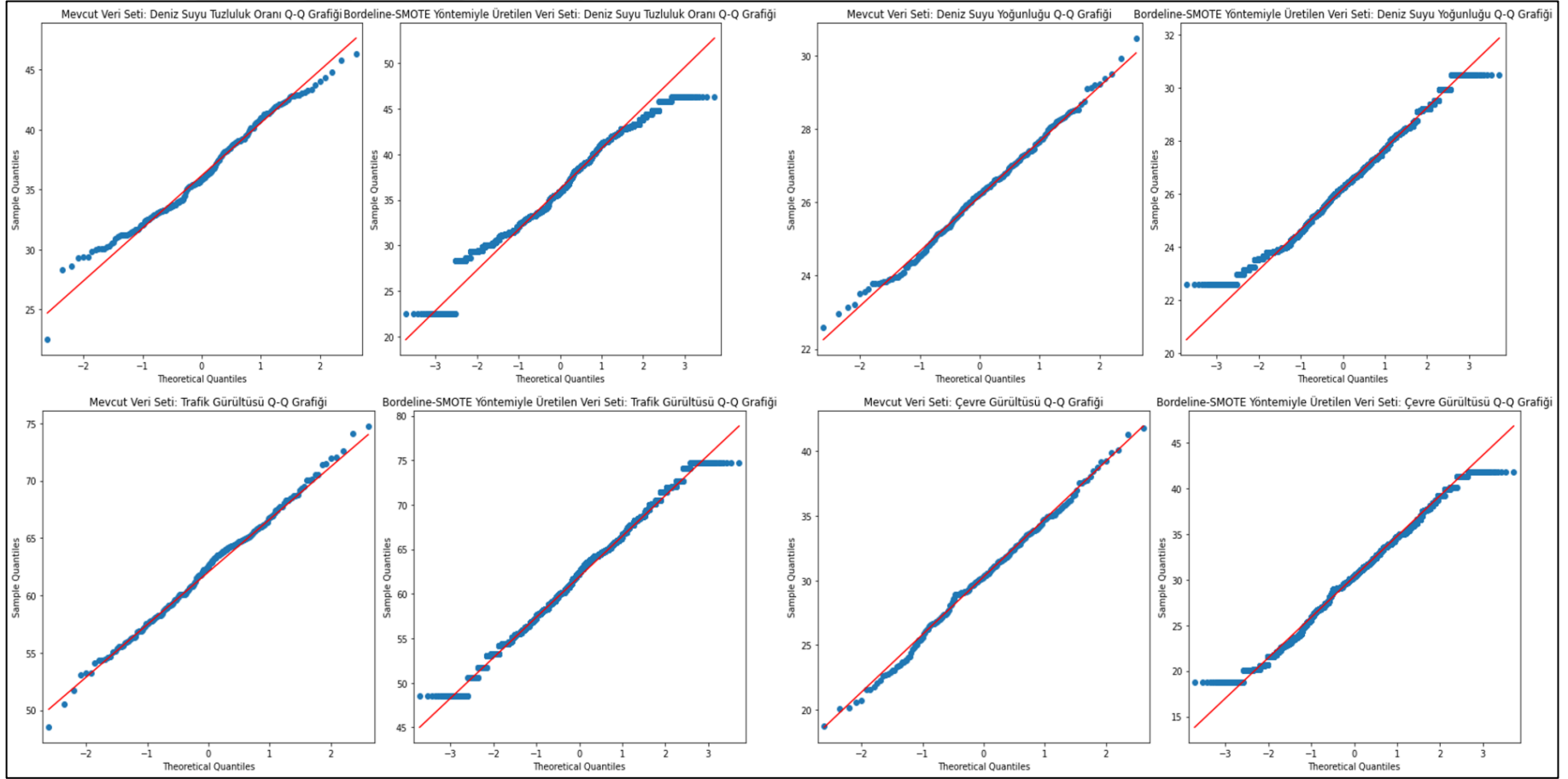
Şekil 22: SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-3.



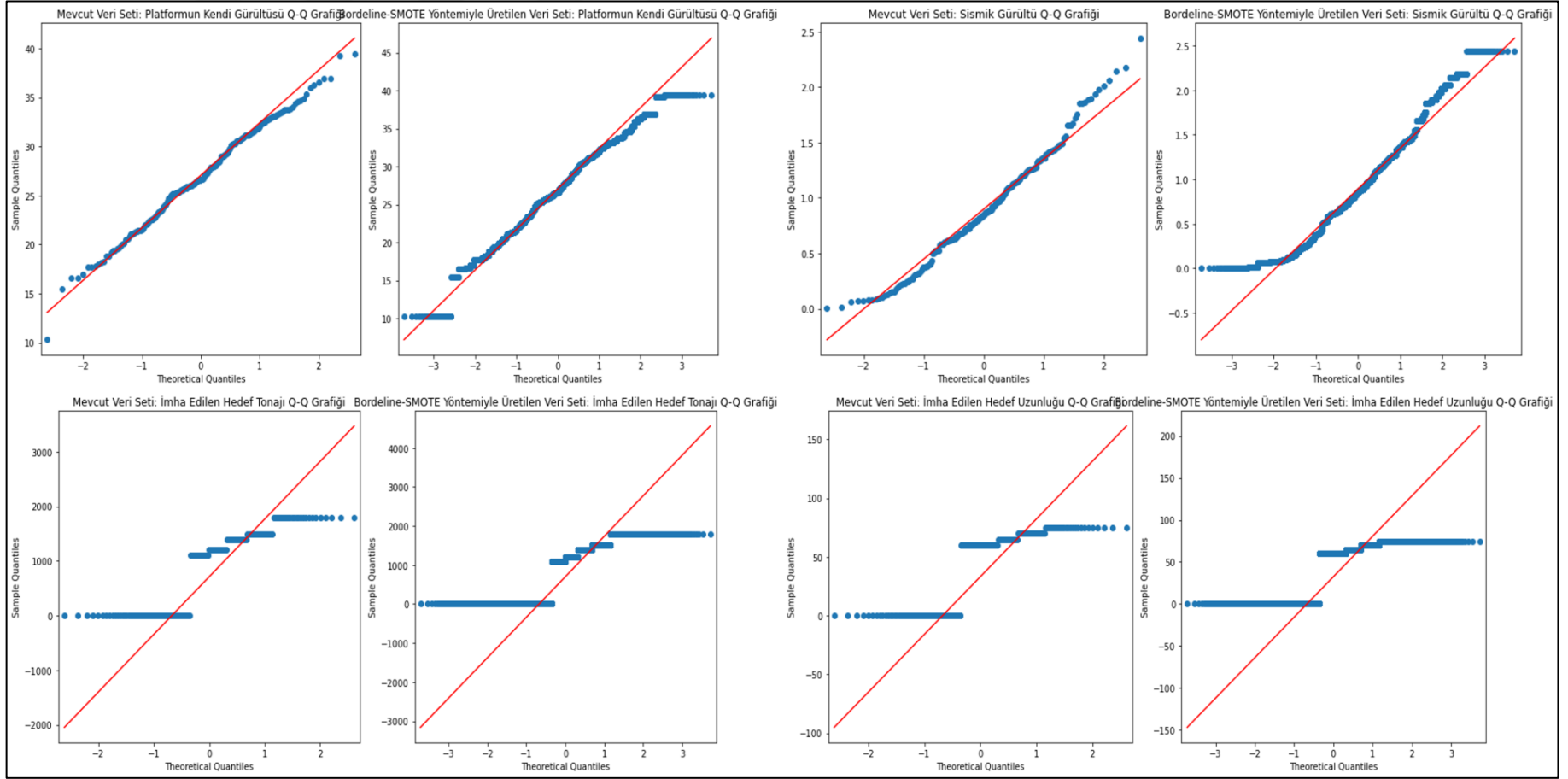
Şekil 23: SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-4.



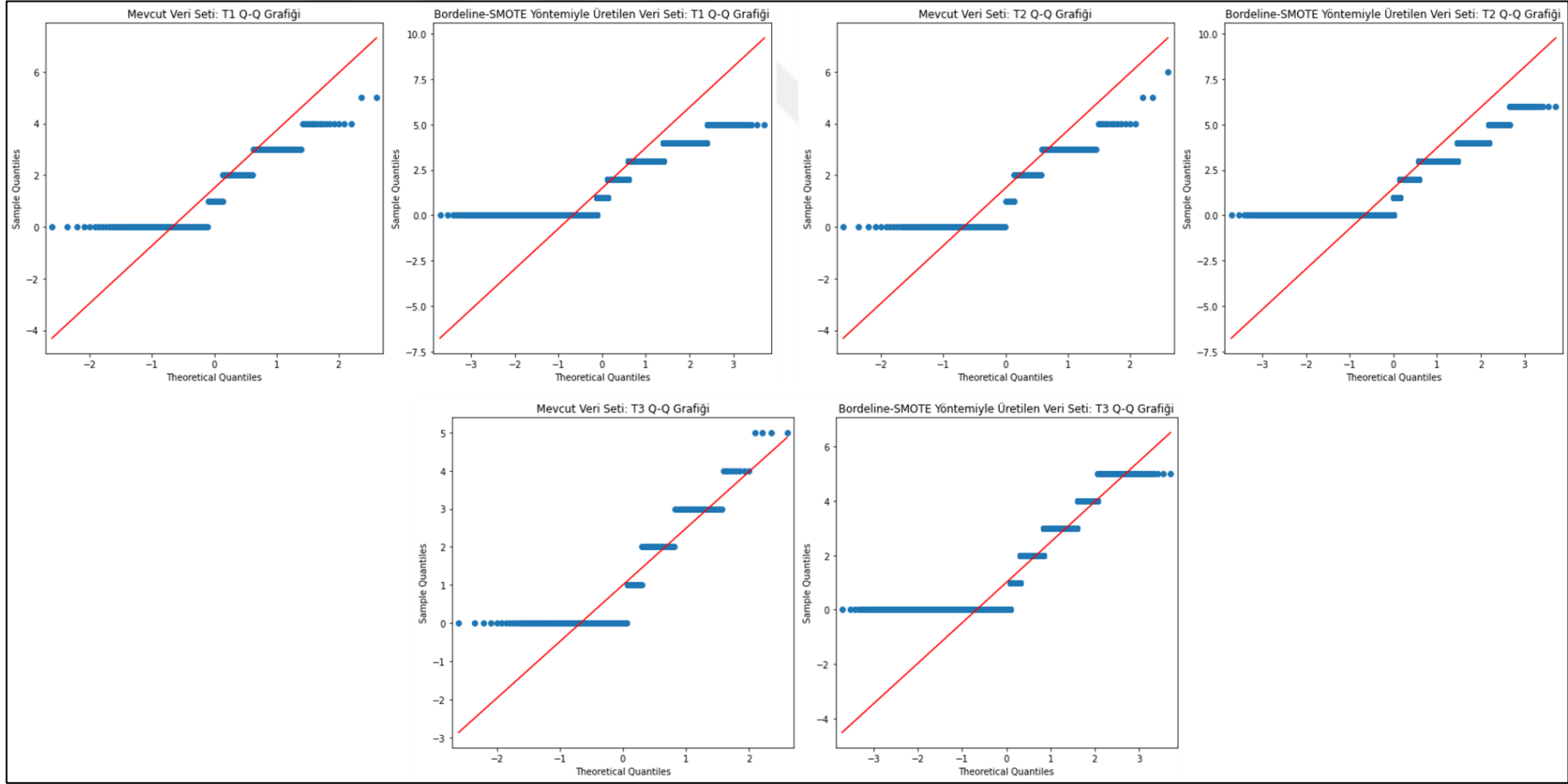
Şekil 24: Borderline SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-1.



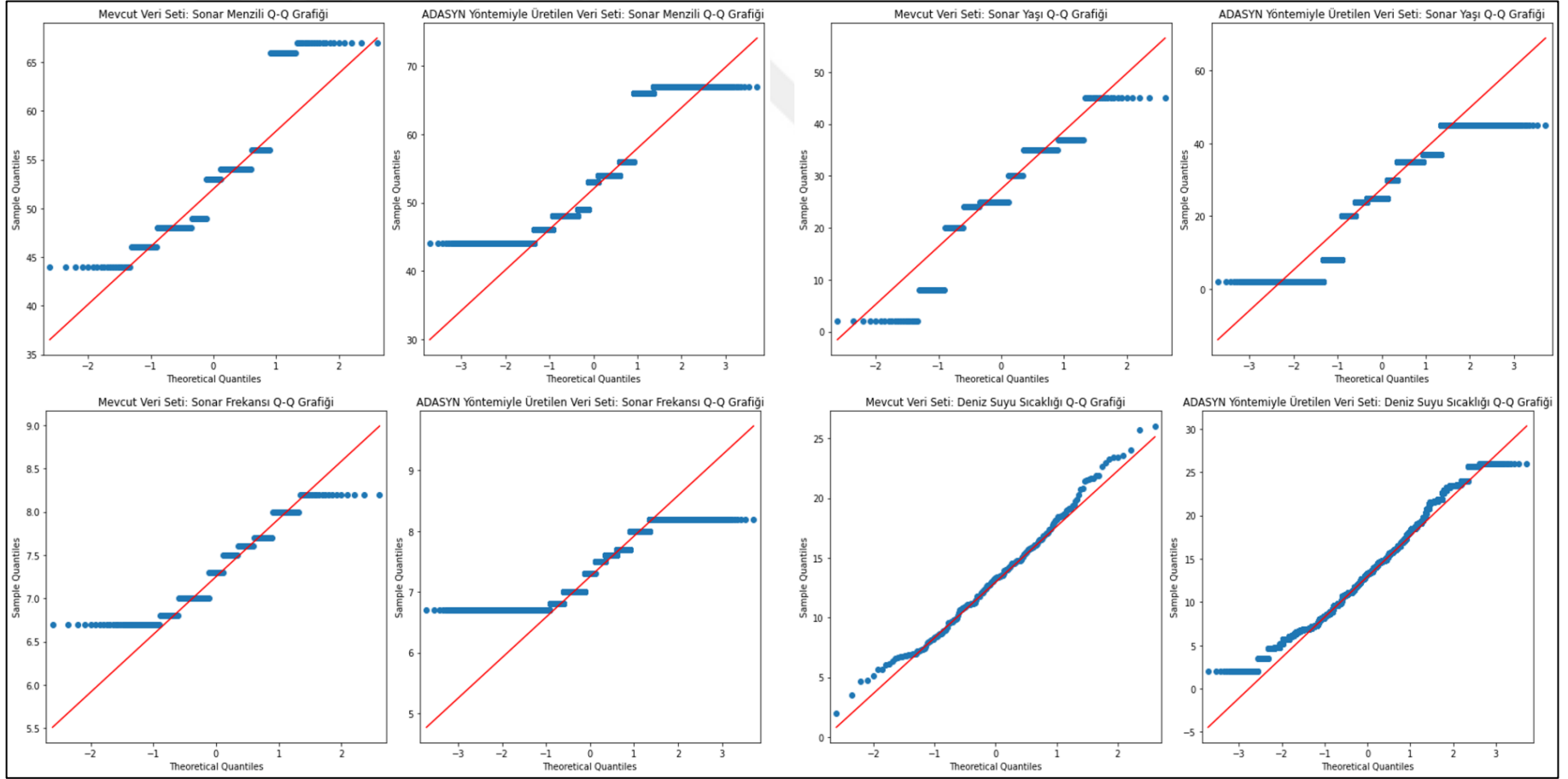
Şekil 25: Borderline SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-2.



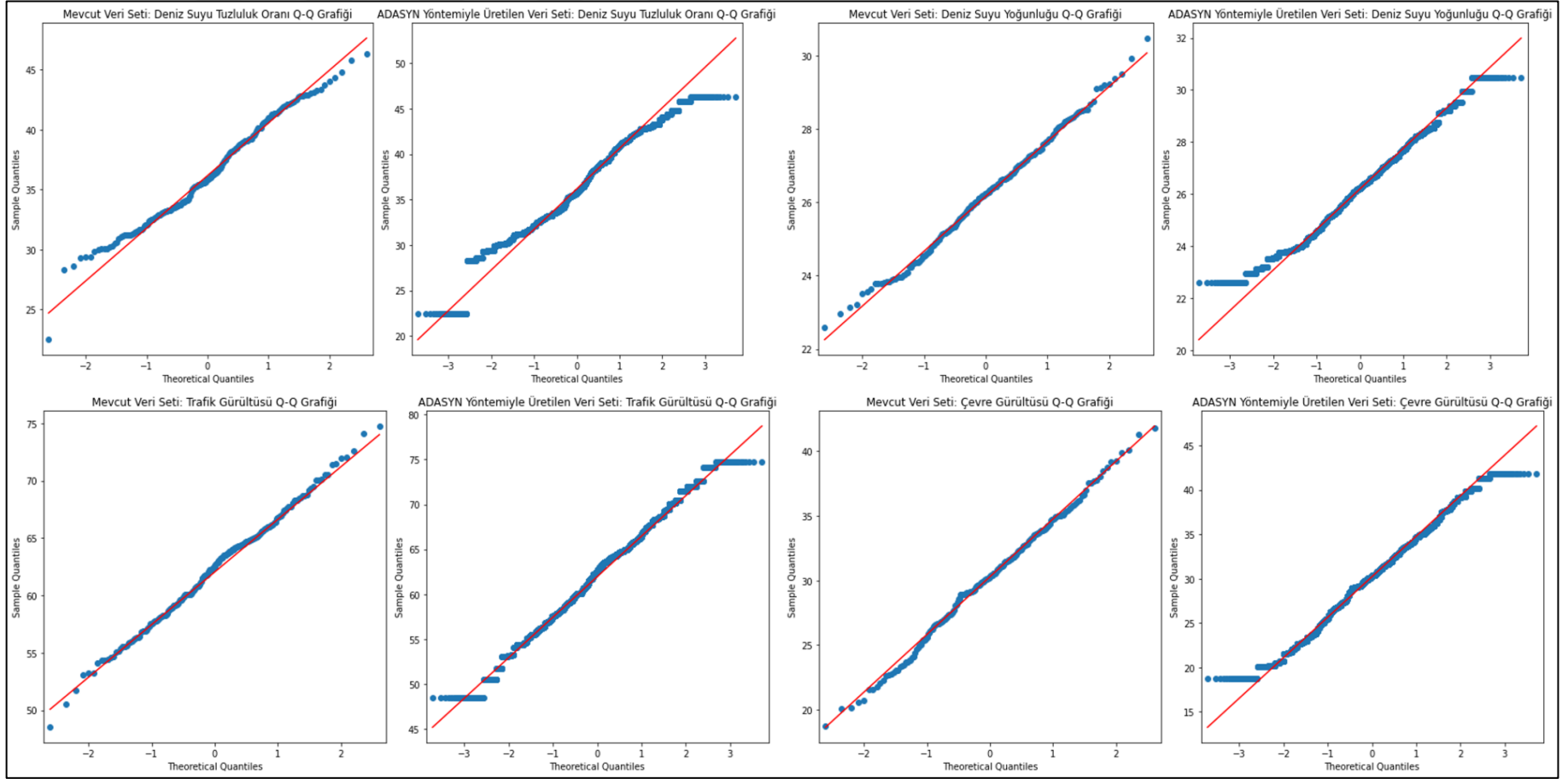
Şekil 26: Borderline SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-3.



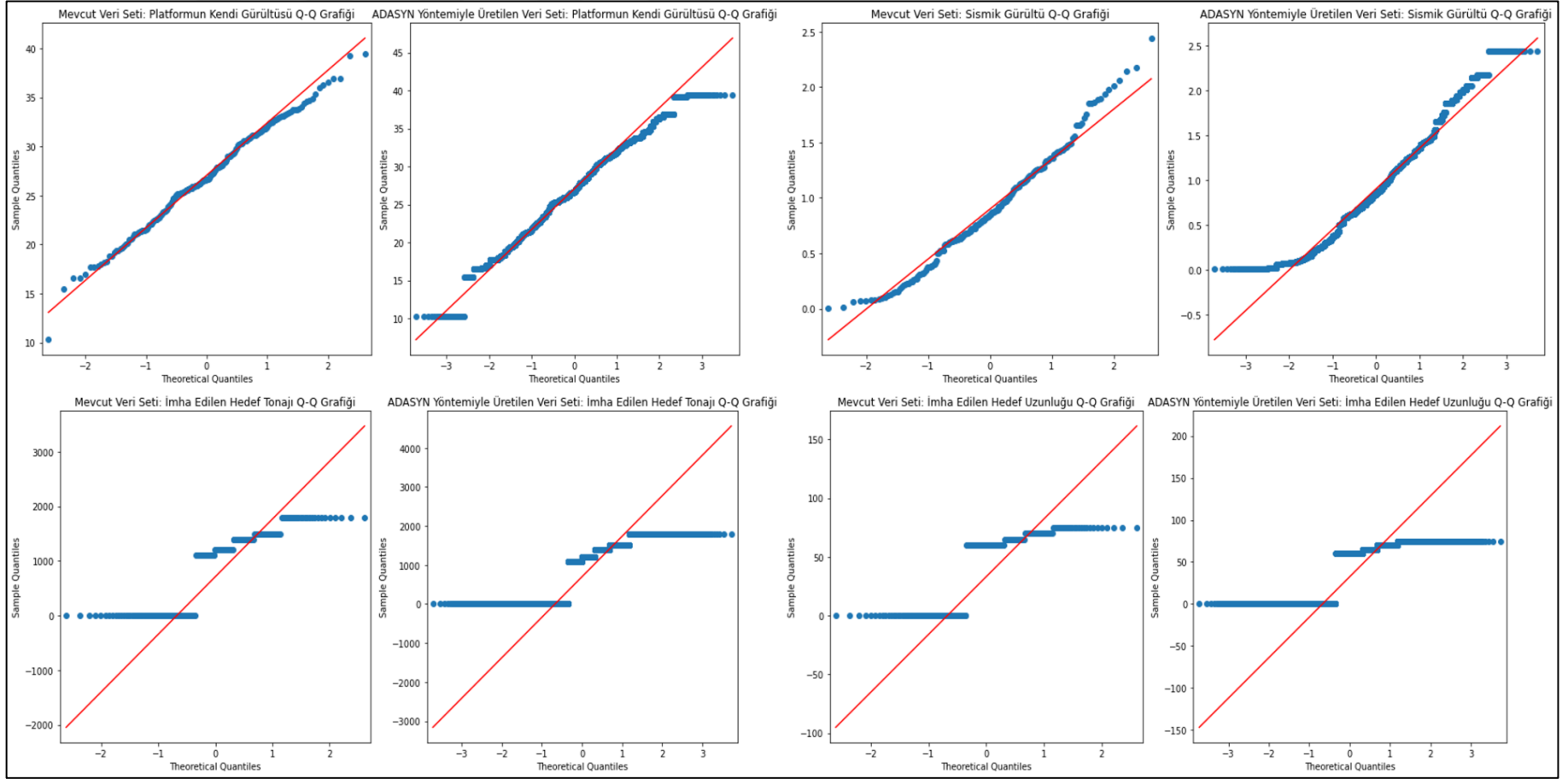
Şekil 27: Borderline SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-4.



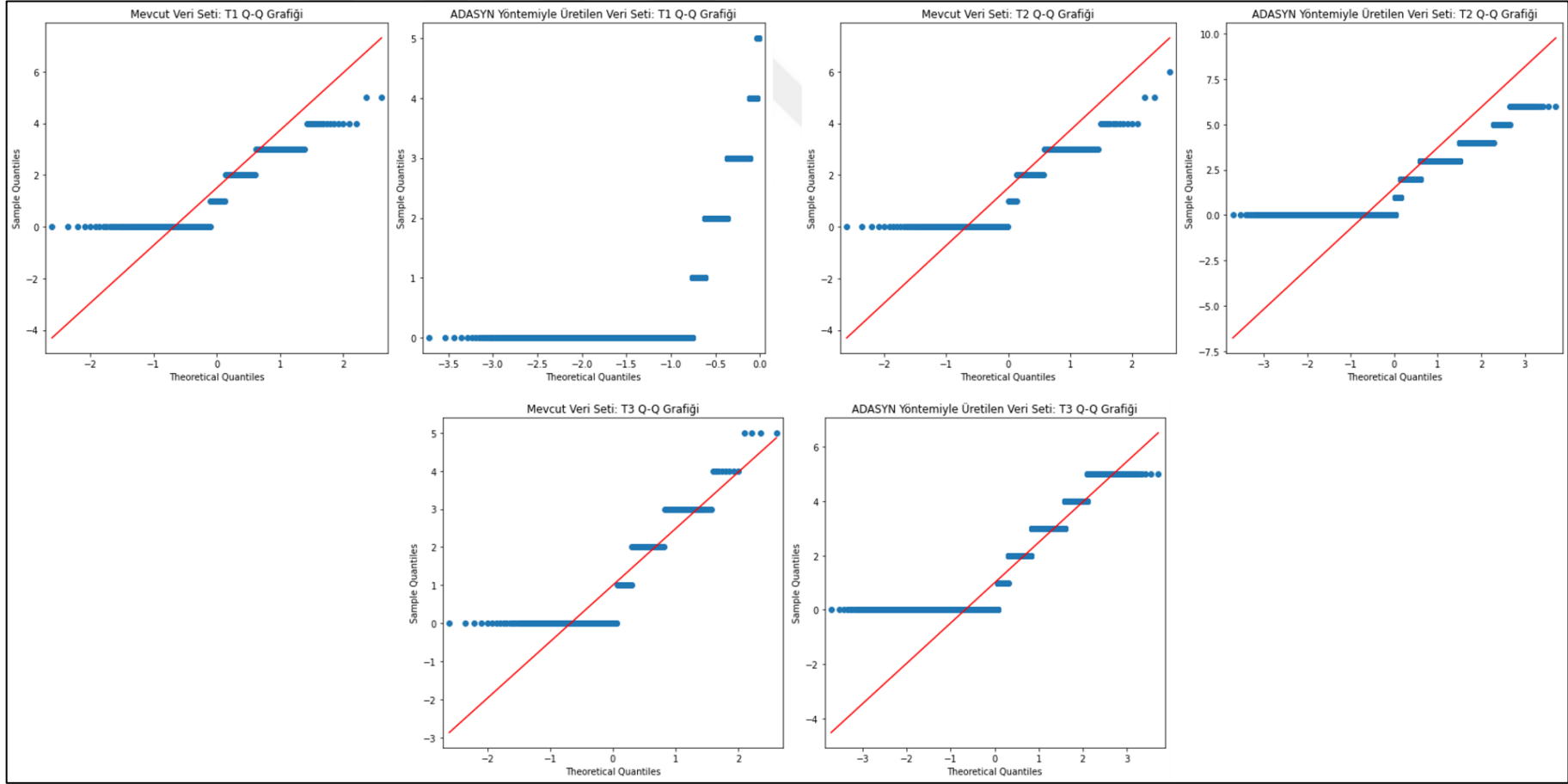
Şekil 28: ADASYN Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-1.



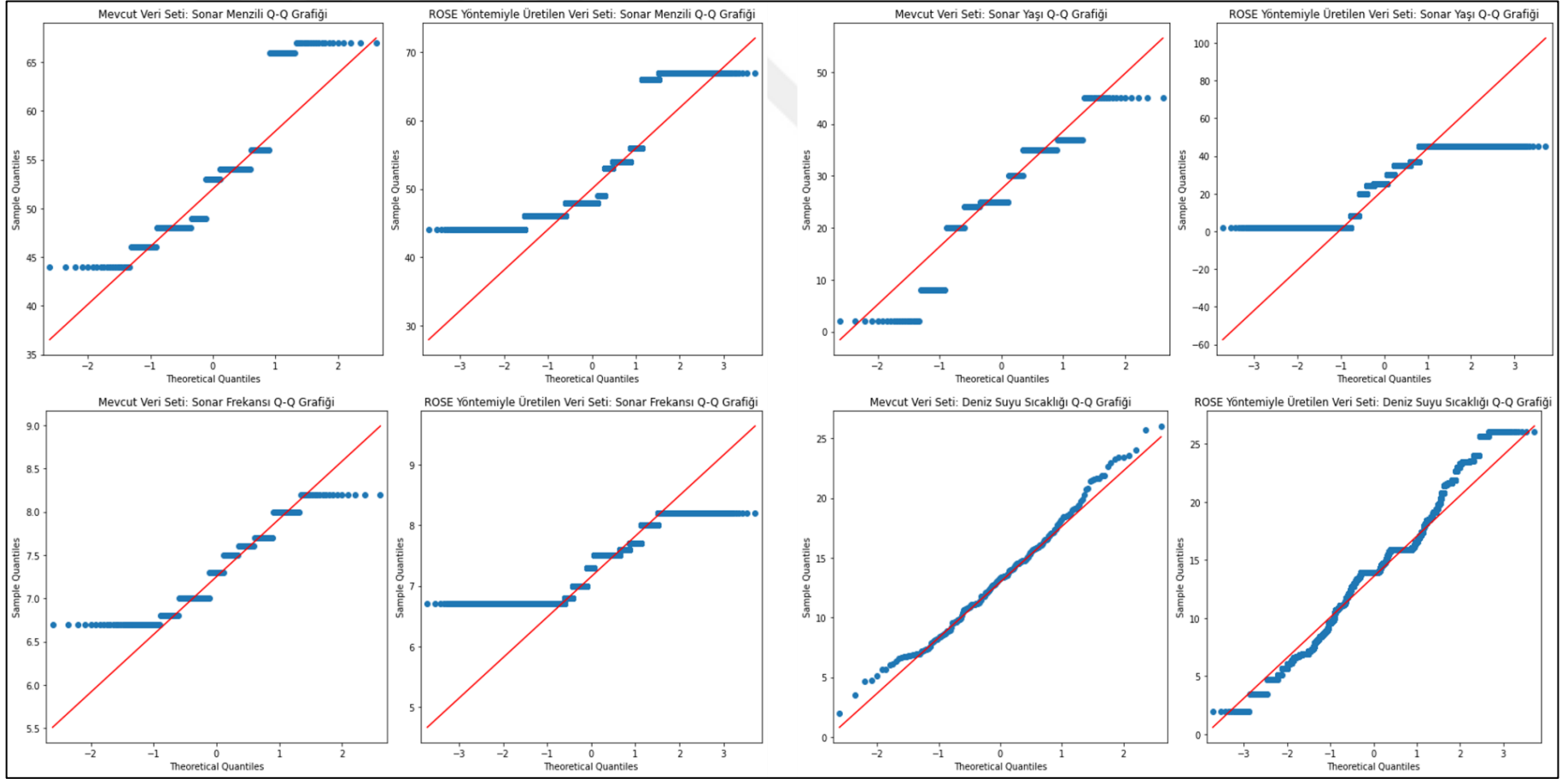
Şekil 29: ADASYN Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-2.



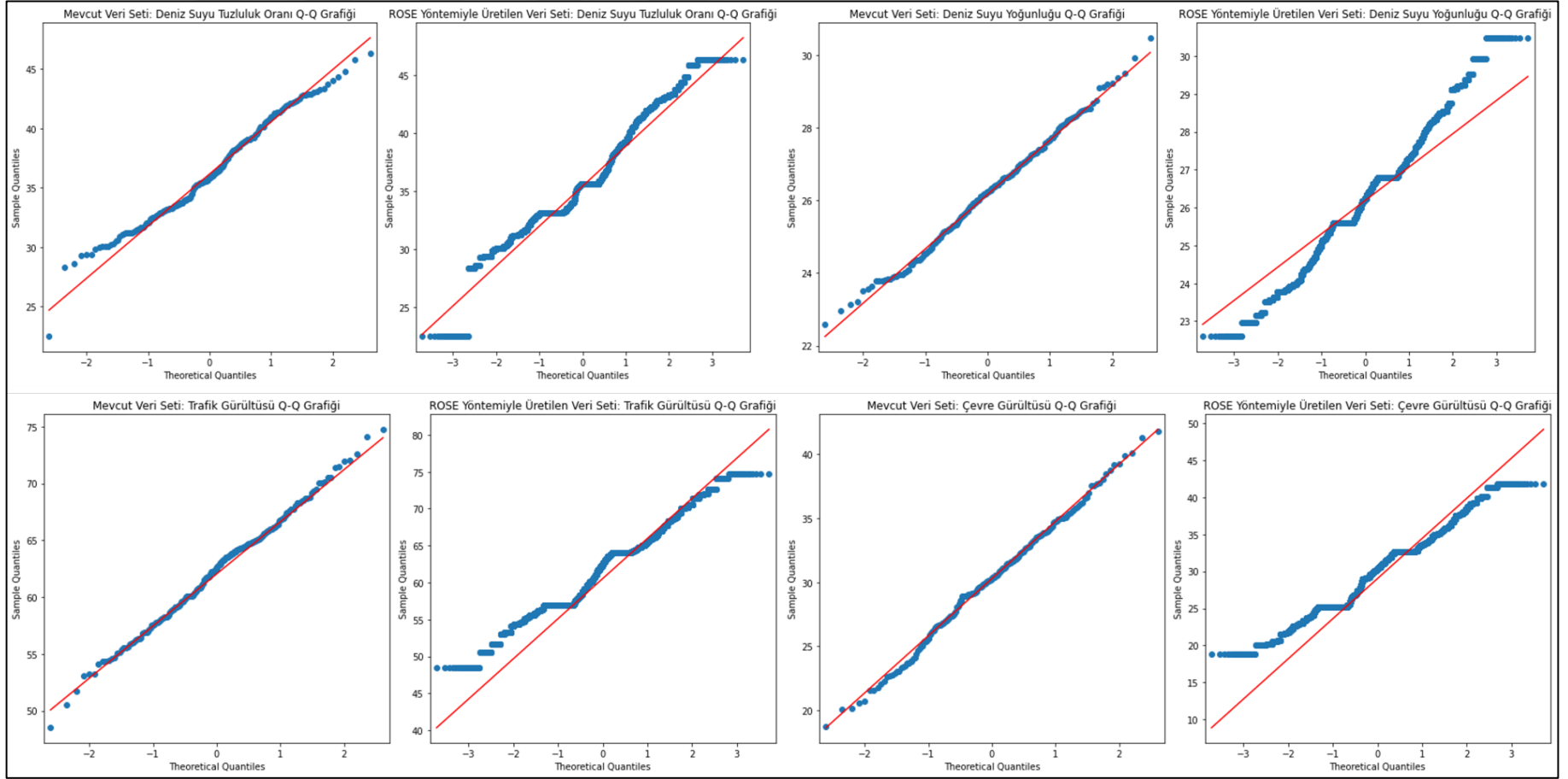
Şekil 30: ADASYN Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-3.



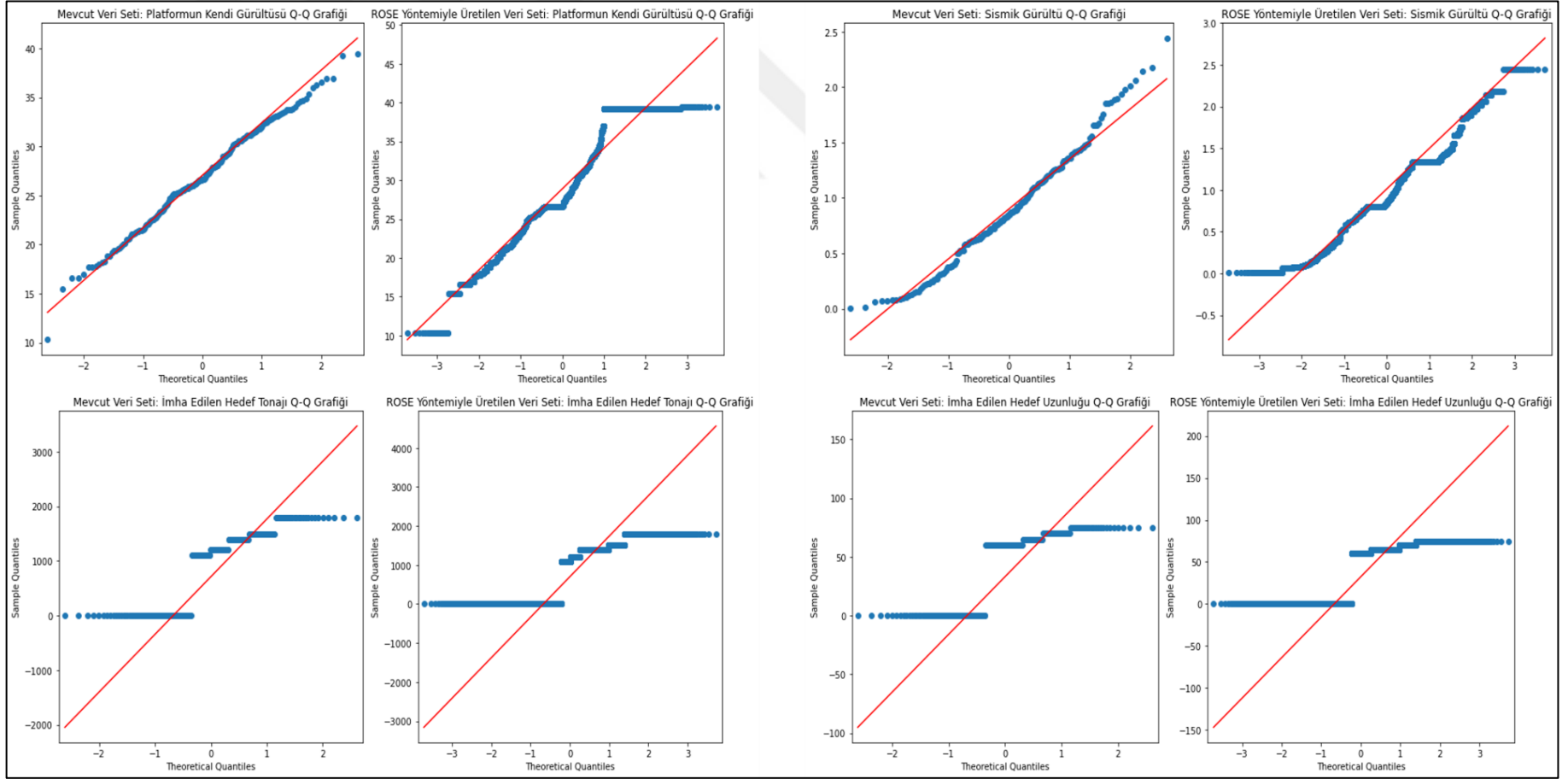
Şekil 31: ADASYN Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-4.



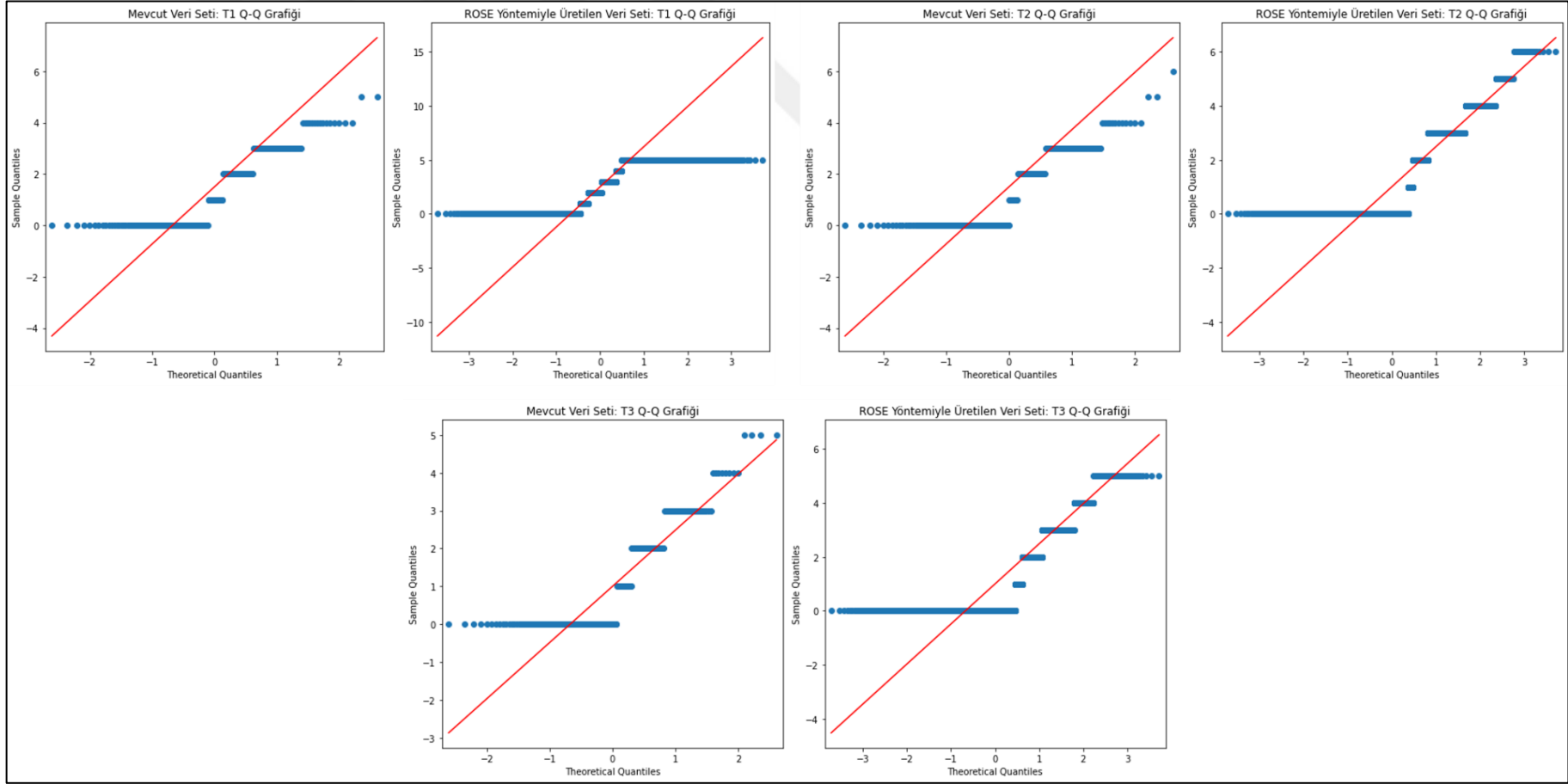
Şekil 32: ROSE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-1.



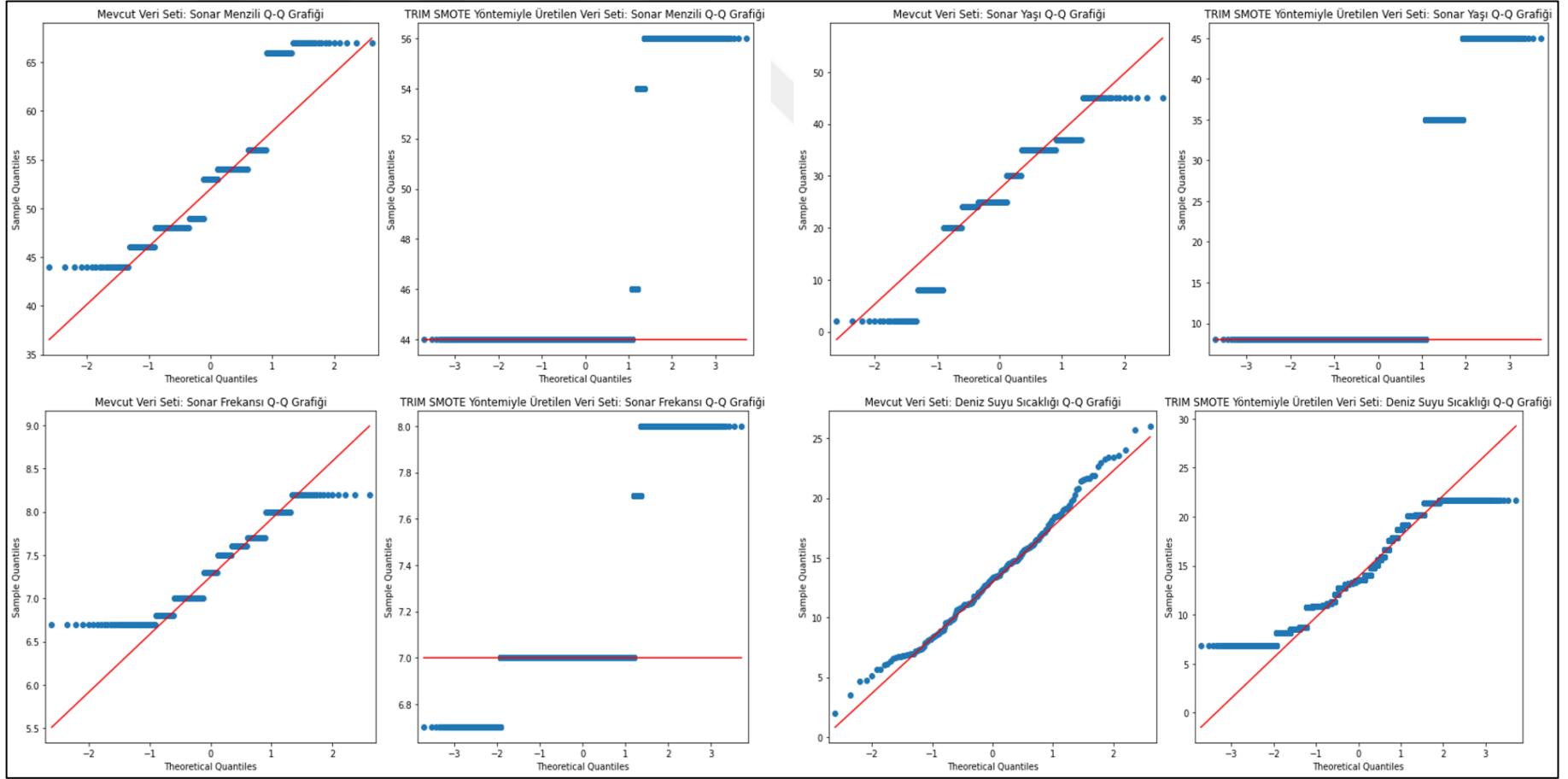
Şekil 33: ROSE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-2.



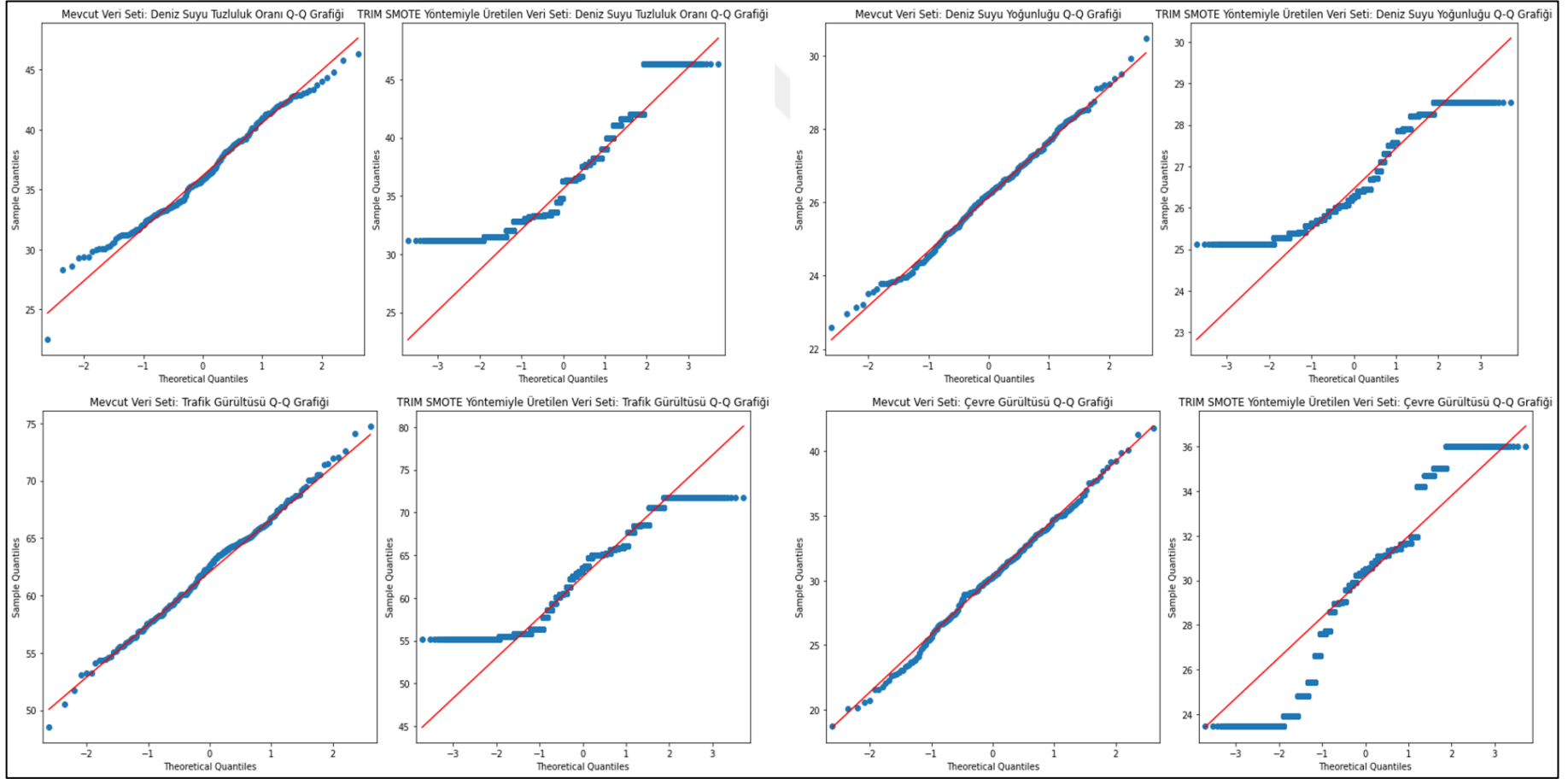
Şekil 34: ROSE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-3.



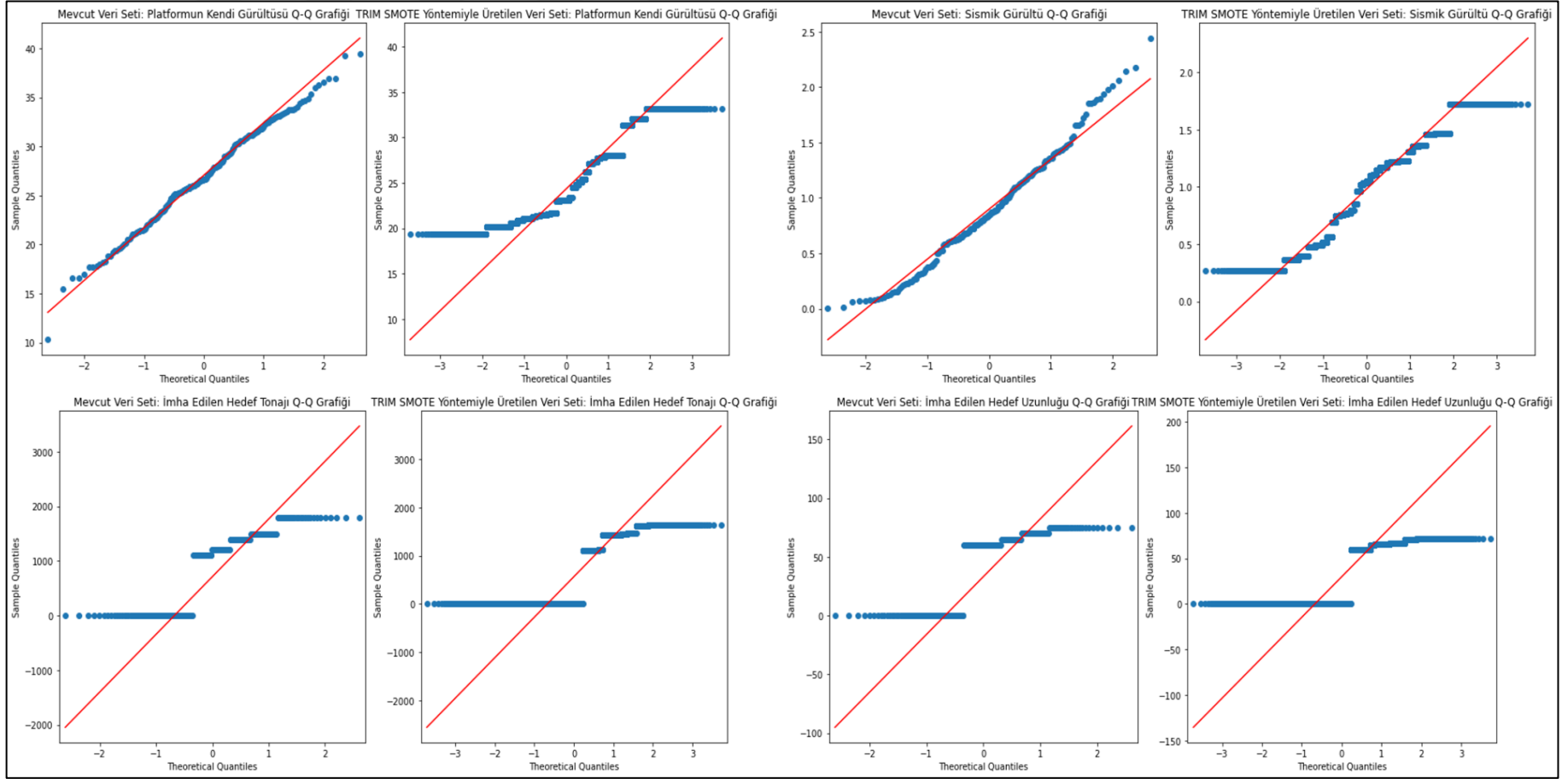
Şekil 35: ROSE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-4.



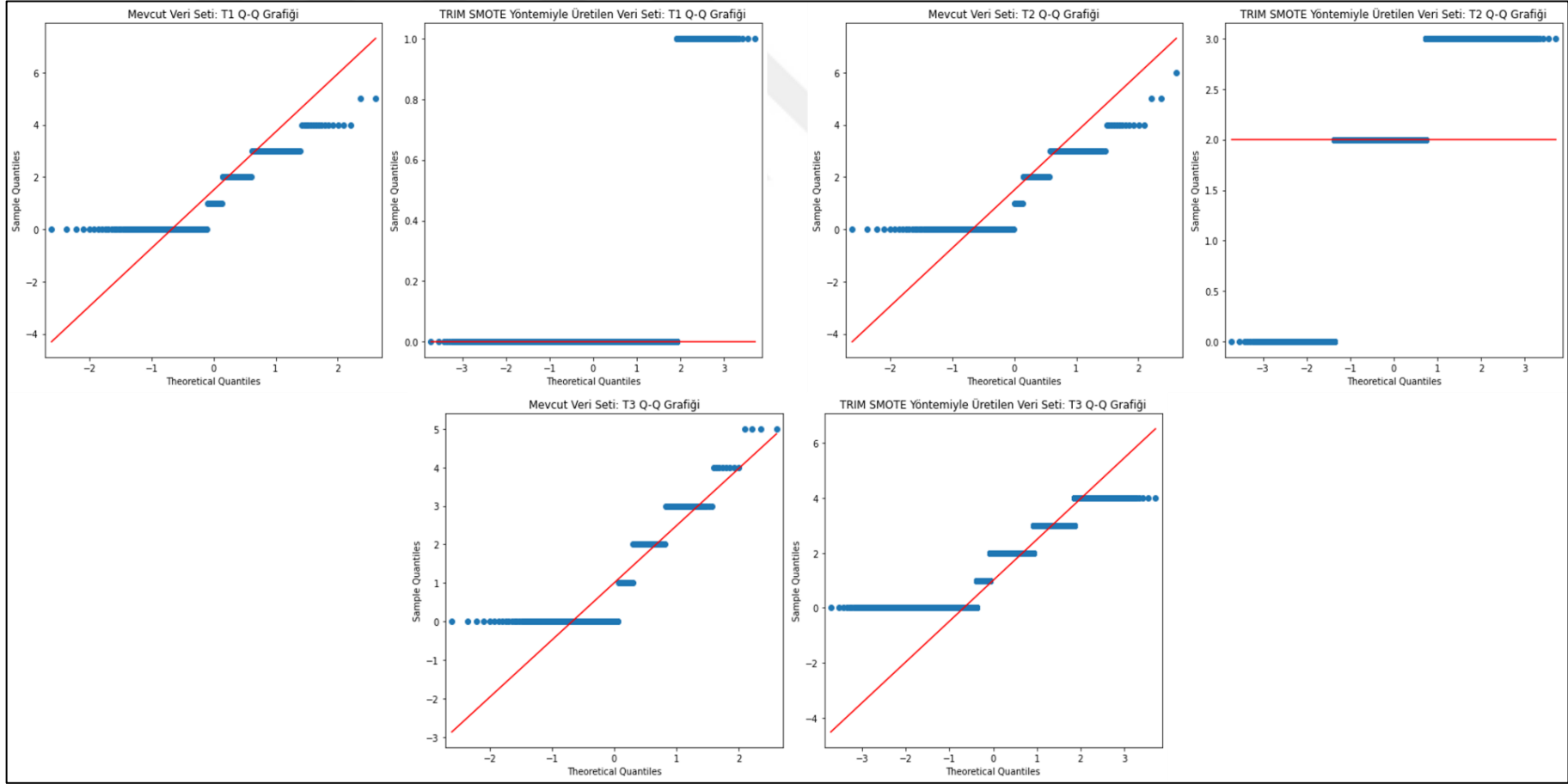
Şekil 36: Trim-SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-1.



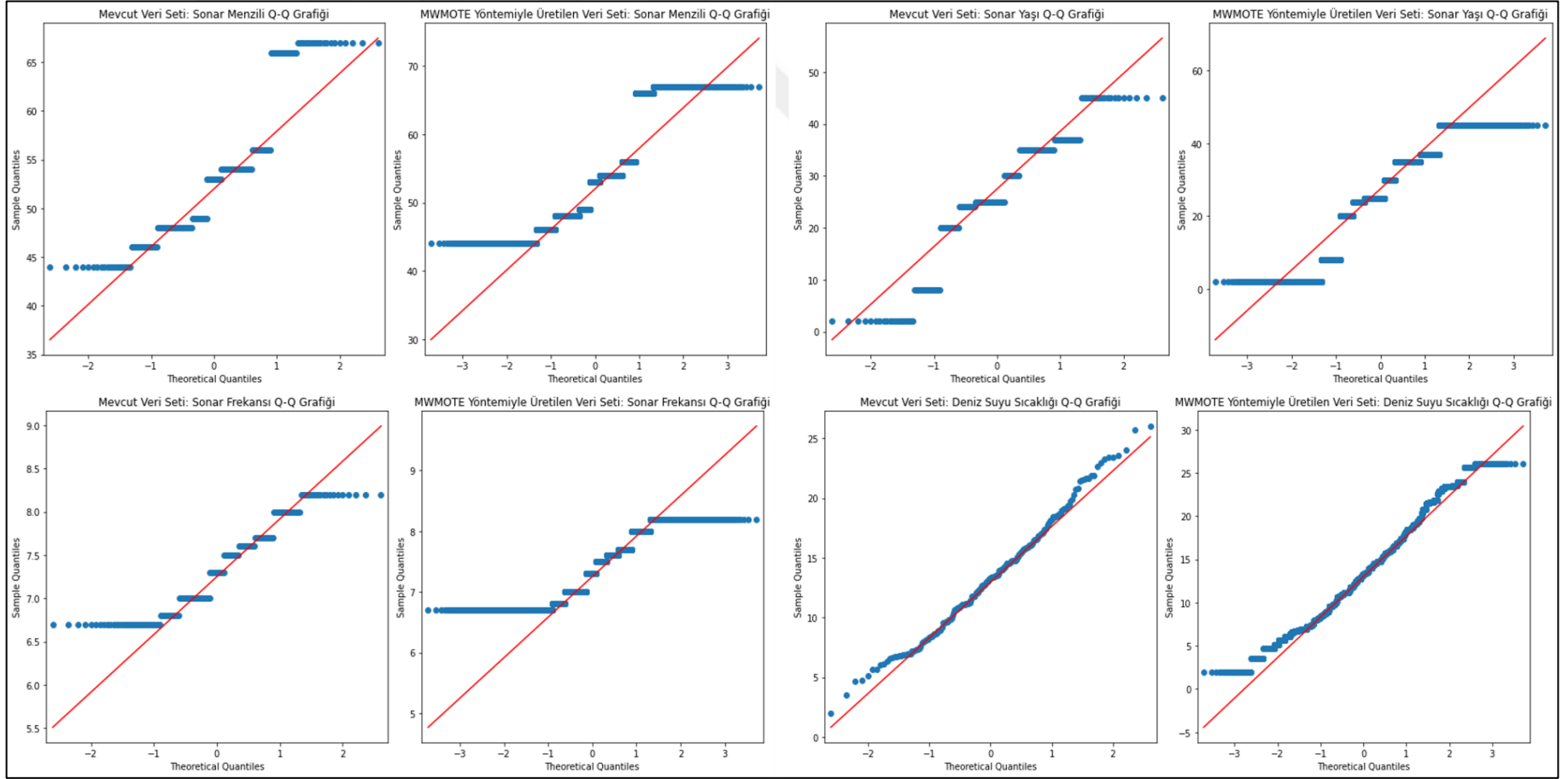
Şekil 37: Trim-SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-2.



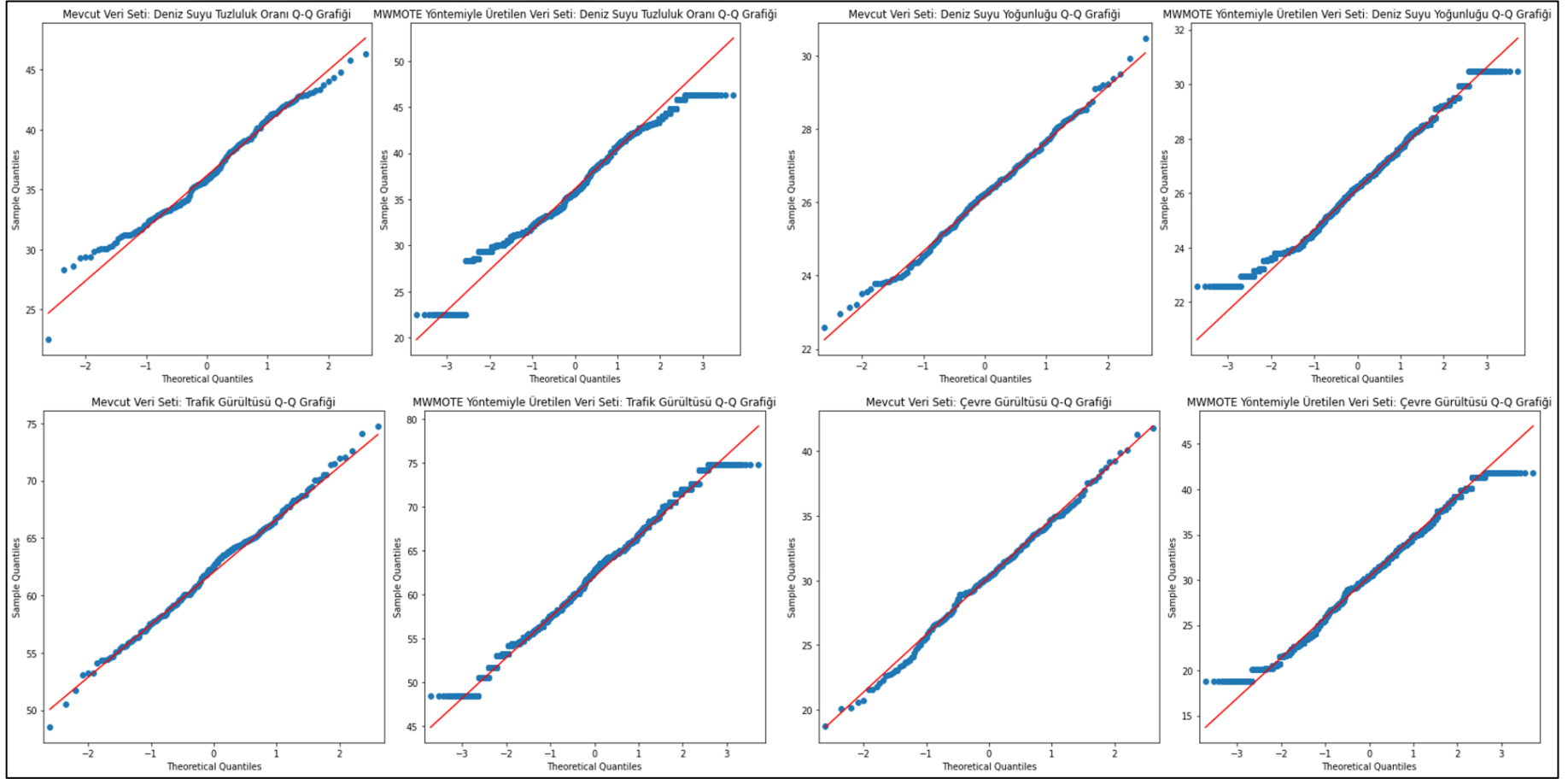
Şekil 38: Trim-SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-3.



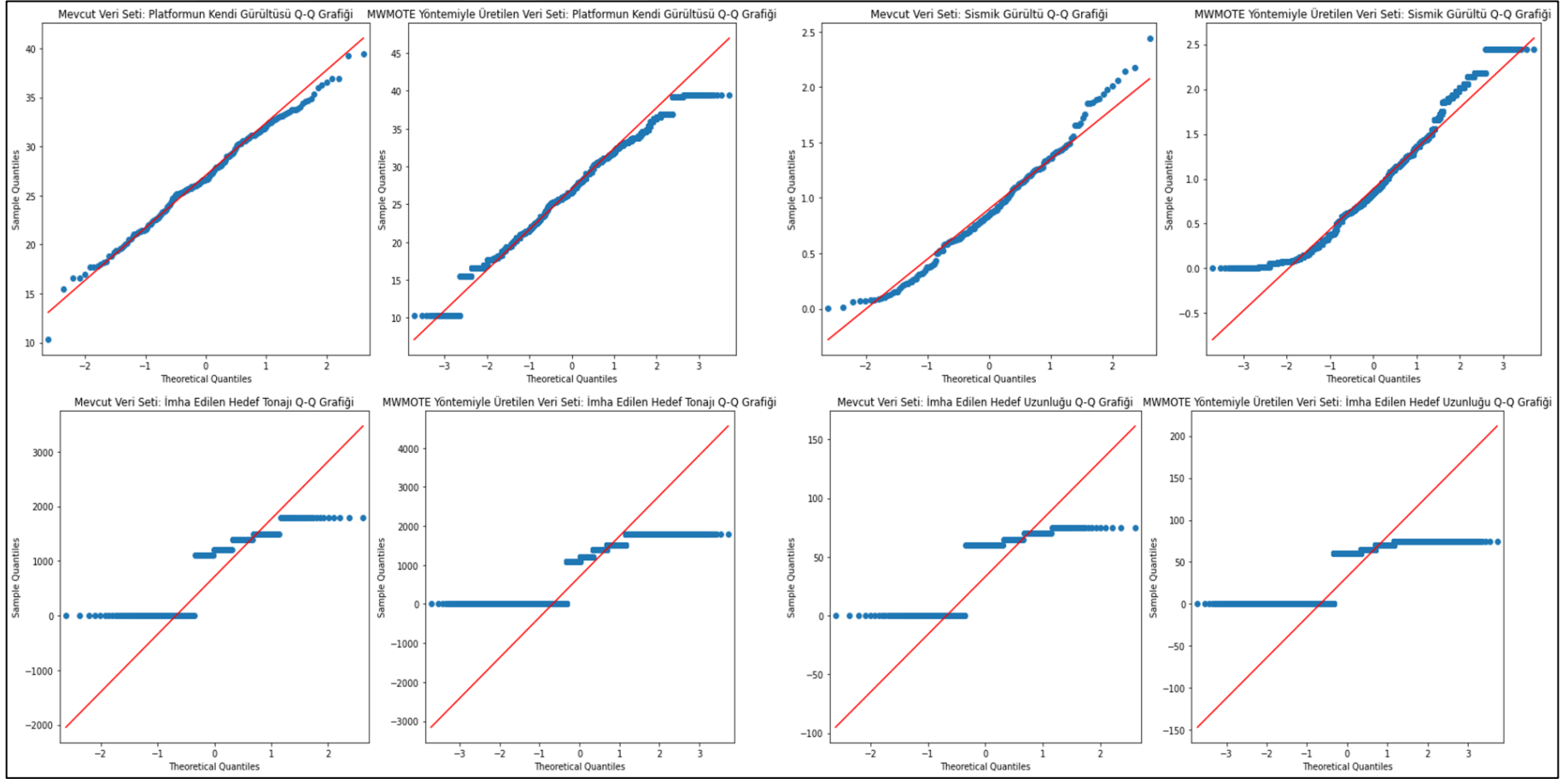
Şekil 39: Trim-SMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-4.



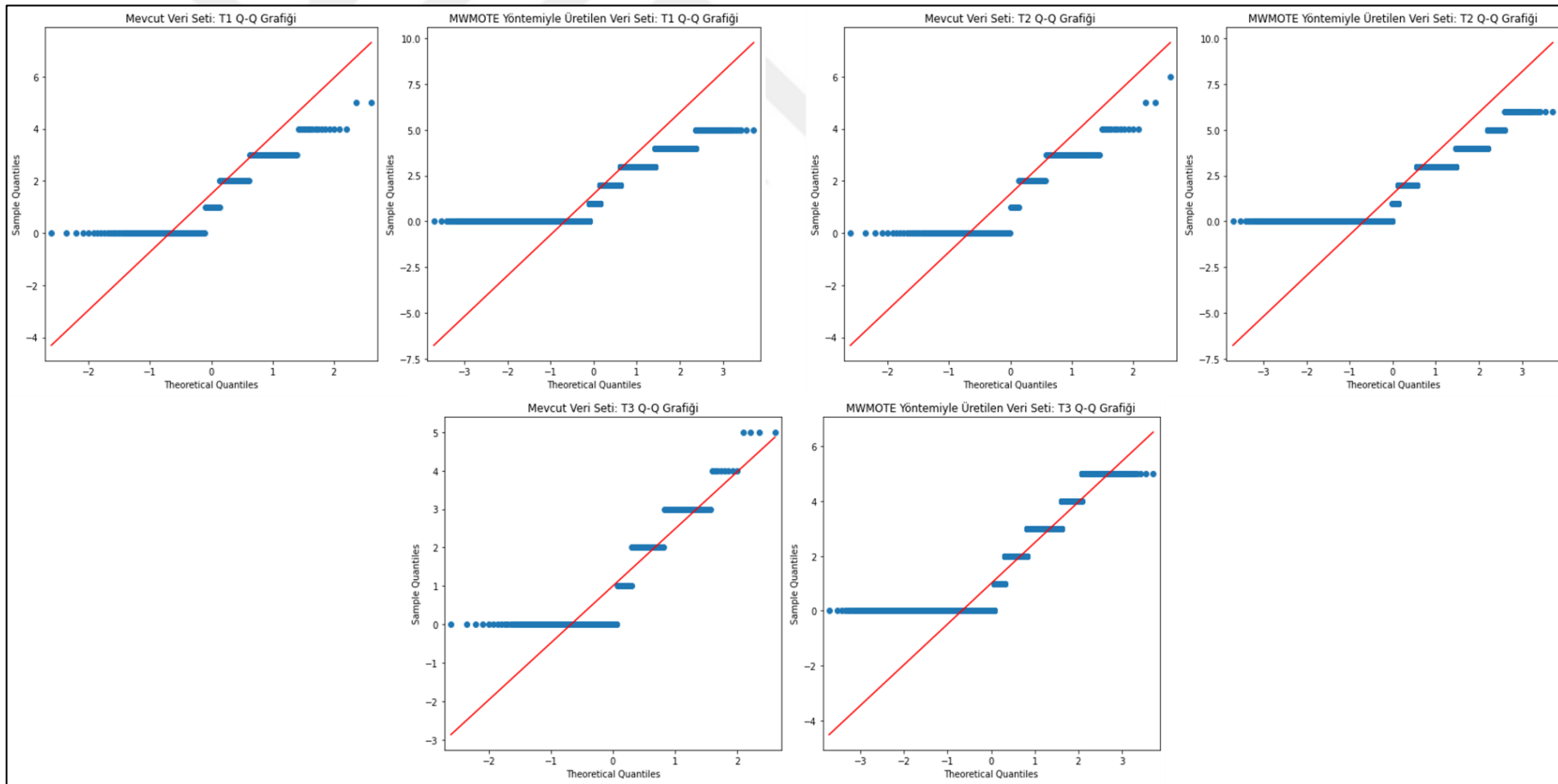
Şekil 40: MWMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-1.

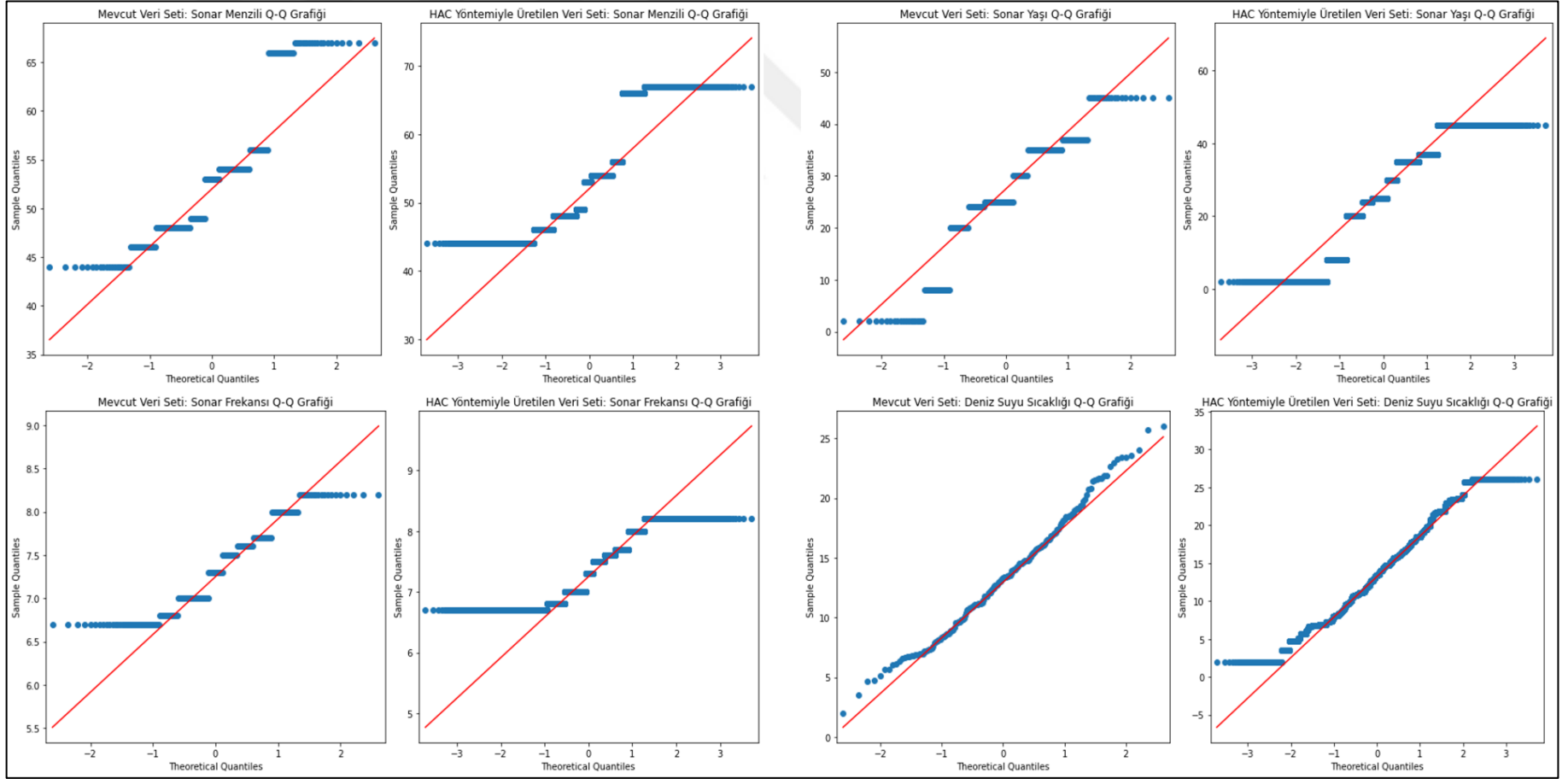


Şekil 41: MWMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-2.

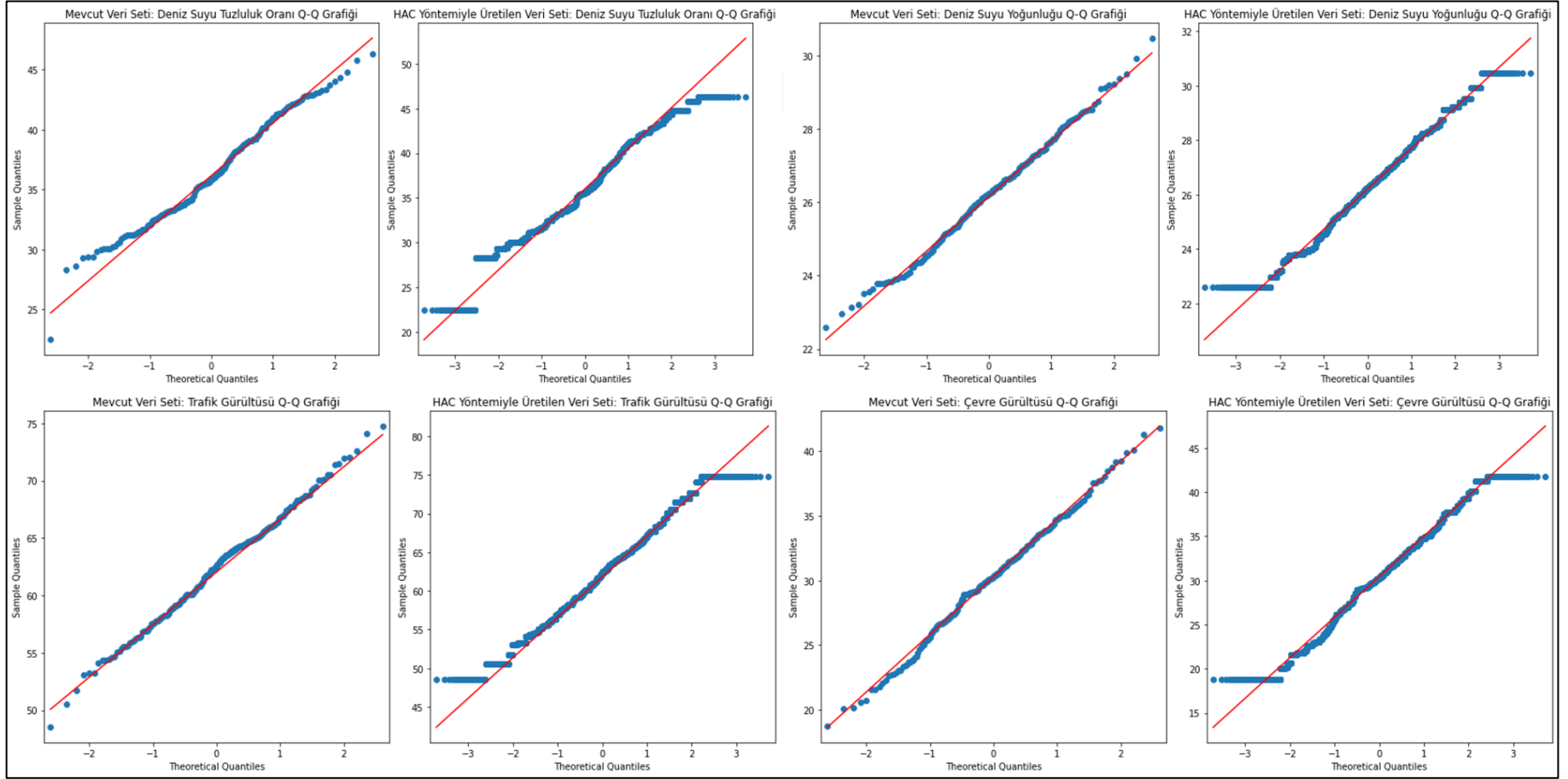


Şekil 42: MWMOTE Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-3.

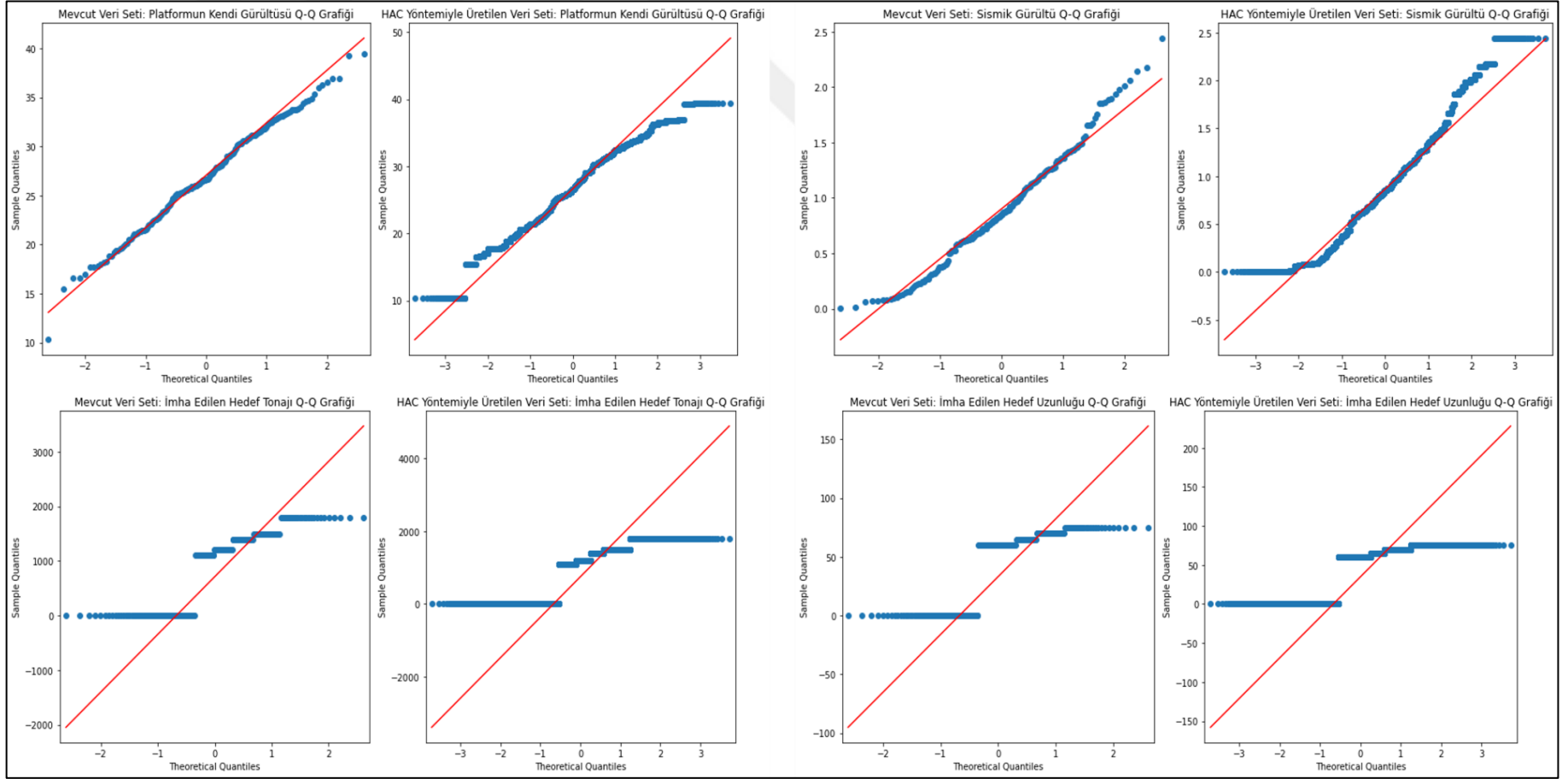




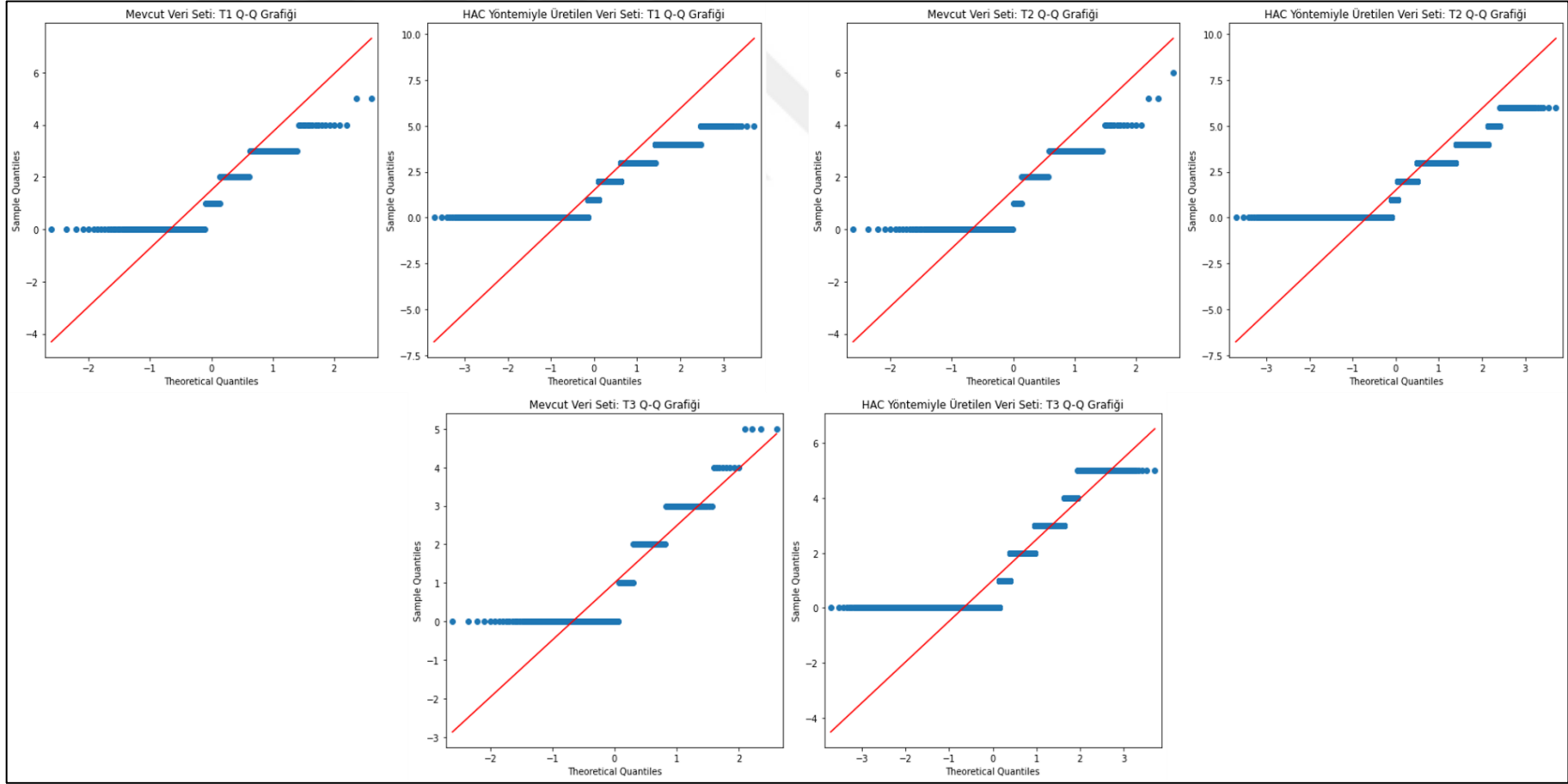
Şekil 44: HAC Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-1.



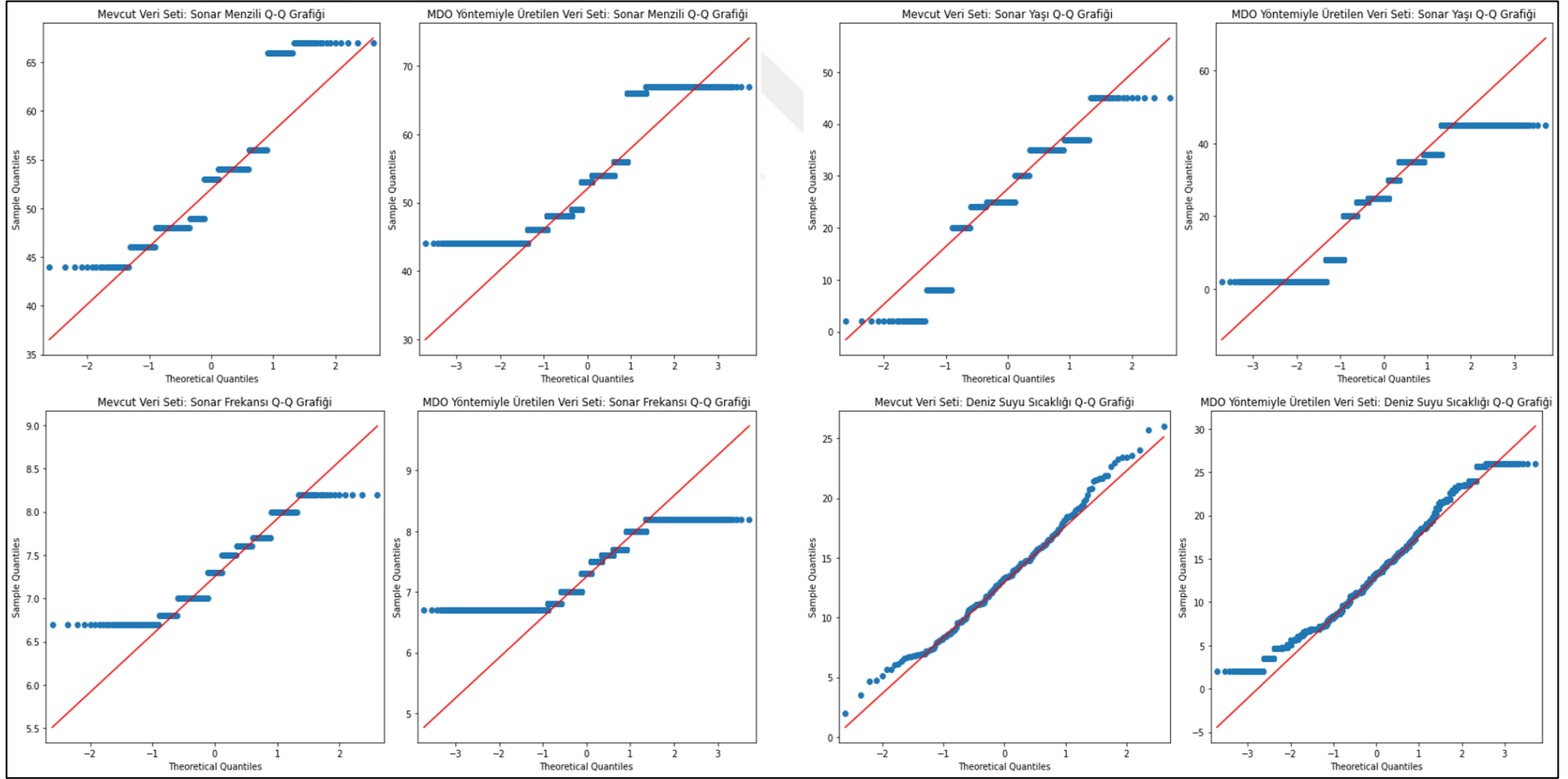
Şekil 45: HAC Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-2.



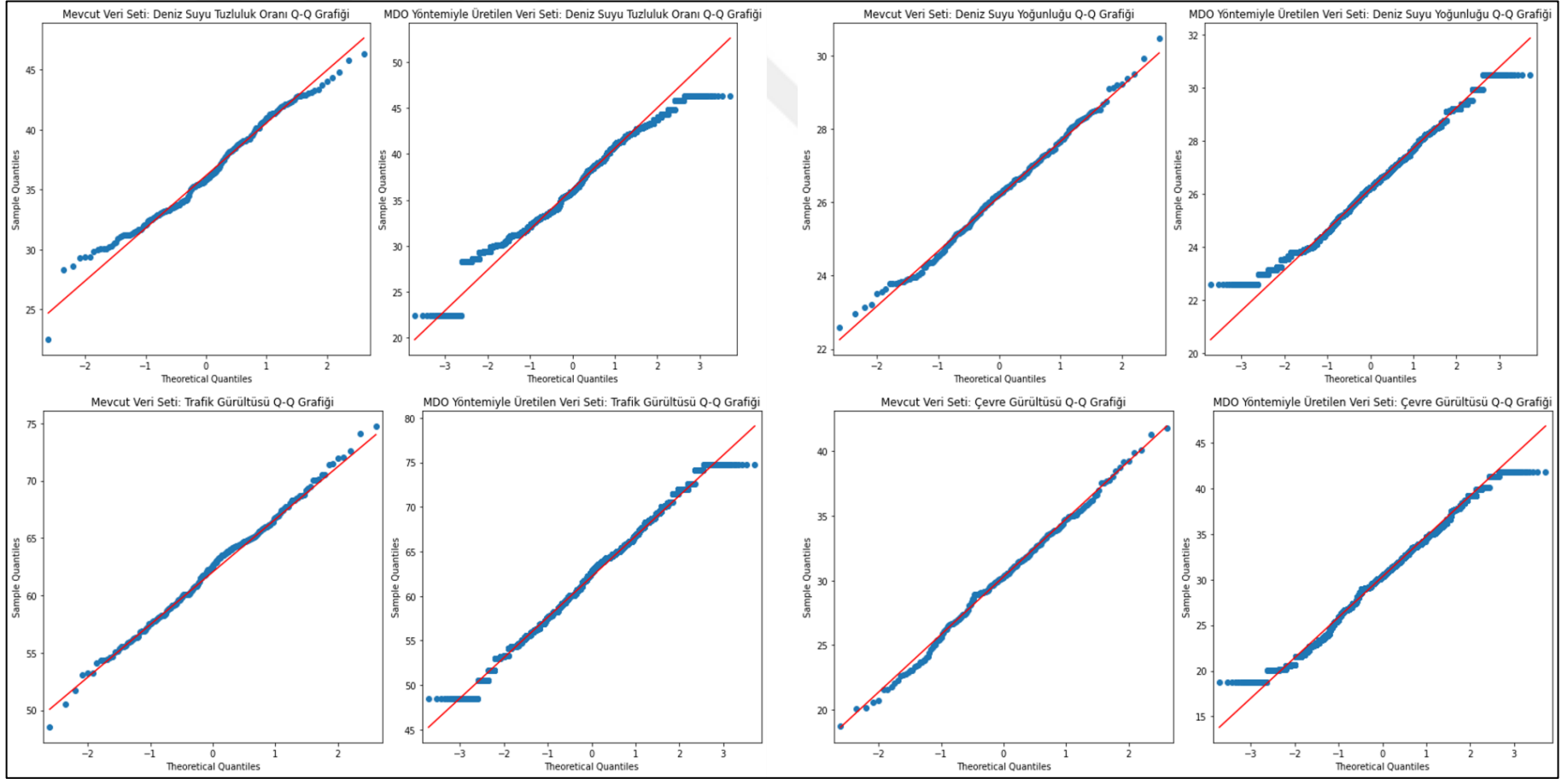
Şekil 46: HAC Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-3.



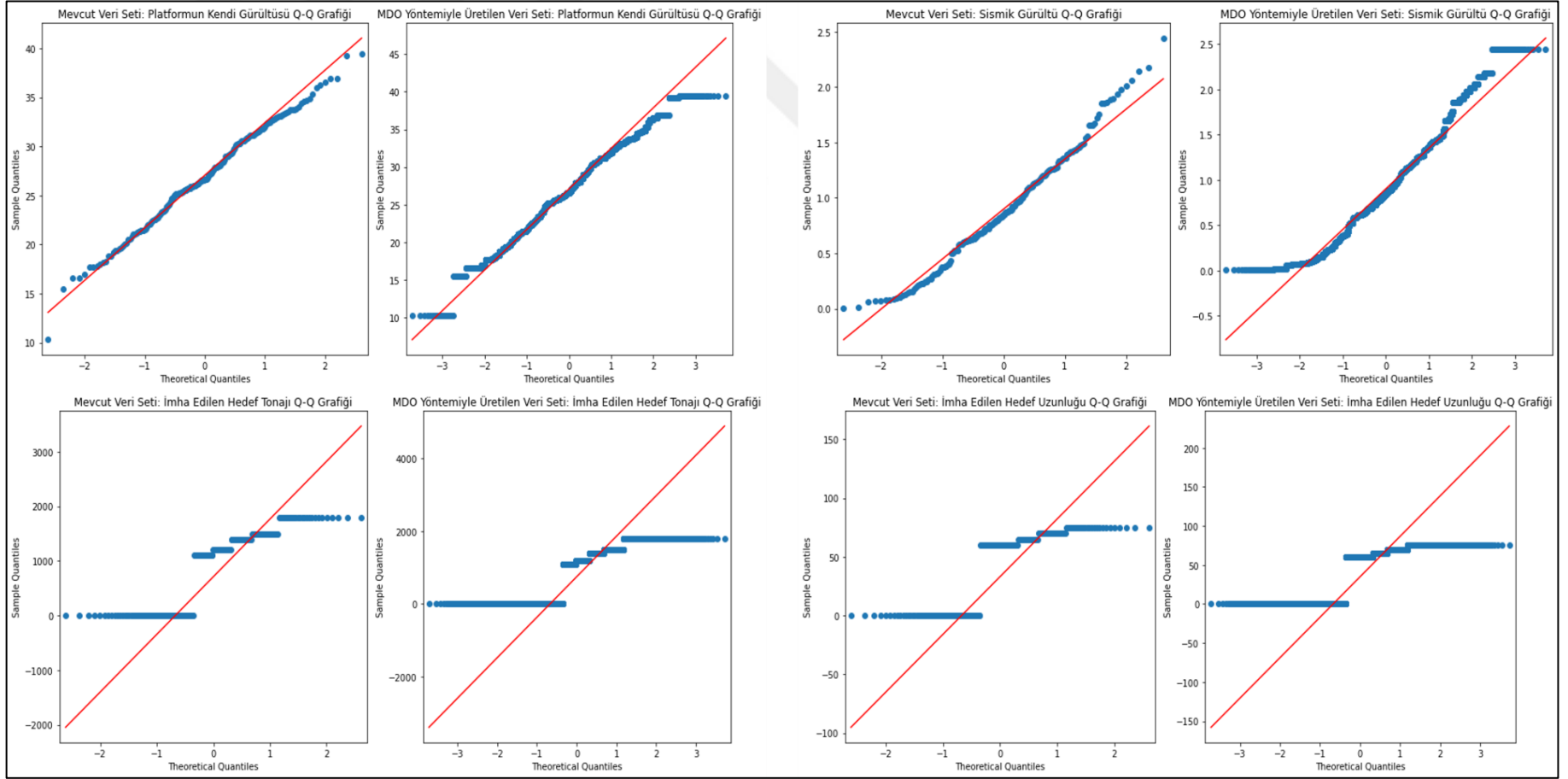
Şekil 47: HAC Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-4.



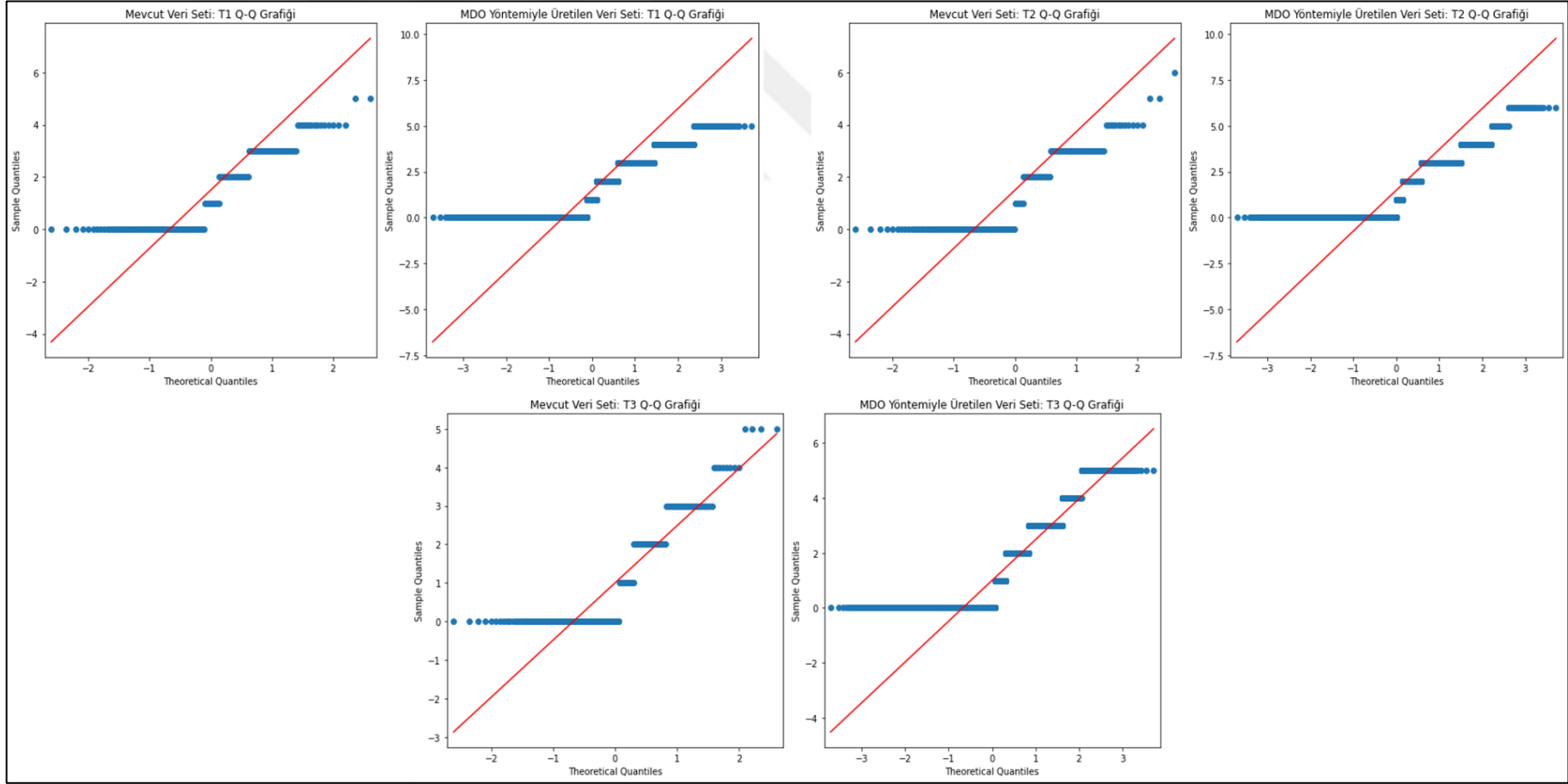
Şekil 48: MDO Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-1.



Şekil 49: MDO Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-2.



Şekil 50: MDO Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-3.



Şekil 51: MDO Yöntemiyle Üretilen Verilerin Q-Q Çizgileri-4.

Şekil (20-51)'de yer alan grafikler incelendiğinde;

- Borderline SMOTE, MWMOTE, HAC ve MDO yöntemleriyle üretilen sentetik verilerin kaynak veri setiyle oldukça benzer yapılara sahip oldukları,
- SMOTE, ADASYN, ROSE ve Trim-SMOTE yöntemleriyle üretilen sentetik verilerin dağılımların ise kaynak veri setiyle oldukça farklı yapılara sahip oldukları görülmektedir.

2.2.2.3 Kolmogorov-Smirnov Testinin Uygulanması

Kolmogorov-Smirnov testi, iki olasılık dağılımının benzerliğini veya bir örnek dağılımının bir teorik dağılıma ne kadar iyi uydurulduğunu değerlendiren istatistiksel bir testtir. Bu test, Kumulatif Dağılım Fonksiyonu (Cumulative Distribution Function, CDF) değerlerini kullanarak ölçüm yapar ve iki dağılım arasındaki en büyük farkı belirler. Söz konusu test, 1933 yılında Andrey Kolmogorov ve Nikolai Smirnov tarafından bağımsız olarak geliştirilmiştir. Temel amacı, bir veri kümesinin belirli bir teorik dağılıma uymasını test etmektir. Test genellikle non-parametrik bir test olarak adlandırılır, çünkü dağılımın parametrelerini bilmek veya varsaymak zorunda değildir. Kolmogorov-Smirnov testi, özellikle küçük örneklem büyüklüklerinde ve parametrik olmayan durumlarda yaygın olarak kullanılır. İki temel formu vardır: tek yönlü (one-sample) ve iki yönlü (two-sample) Kolmogorov-Smirnov testleri. Tek yönlü test, bir örnek dağılımının belirli bir teorik dağılıma uymasını sınar; iki yönlü test ise iki örnek dağılım arasındaki benzerliği değerlendirir. Kolmogorov-Smirnov testi, özellikle örnek dağılımının teorik bir dağılıma uygunluğunu belirlemek ve iki örnek arasındaki benzerliği sınamak için istatistiksel analizlerde yaygın olarak kullanılan güçlü bir araçtır [48].

Kolmogorov-Smirnov testinde, p değeri (KS istatistiğine dayalı olarak hesaplanan bir olasılık değeri) alfa seviyesiyle karşılaştırılarak istatistiksel anlamlılık belirlenir. Alfa seviyesi, kabul edilebilir bir hata seviyesini temsil eder ve genellikle 0.05 olarak kullanılır. Bu, %5 anlam düzeyine karşılık gelir ve p değeri 0.05'ten küçükse, null hipotezi reddedilir ve iki örnek dağılımının aynı olmadığı sonucuna varılır.

Dolayısıyla, bu çalışmada da alpha değeri 0,05 olarak alınmış ve p değeri:

- $p < \alpha$ ise: İki örnek dağılım arasında istatistiksel olarak anlamlı bir fark vardır.

• $p \geq \alpha$ ise: İki örnek dağılım arasında istatistiksel olarak anlamlı bir fark yoktur. Hipotezlerine göre her bir sentetik veri seti ile kaynak veri seti arasında anlamlı bir fark olup olmadığının test edilmiştir. Yapılan hesaplamalara ilişkin sonuçlar Tablo 7’de sunulmuştur.

Tablo 7: Kolmogorov Simirnov Testi Sonuçları.

Veri Seti Adı	Sütun Adı	<i>p value</i> Değeri	Farklılık Durumu
SMOTE Yöntemiyle Üretilen Sentetik Veri	Sonar Menzili	$9,85 \times 10^{-19}$	Anlamlı Bir Fark Vardır
	Sonar Yaşı	$5,46 \times 10^{-13}$	
	Sonar Frekansı	$7,34 \times 10^{-25}$	
	Deniz Suyu Sıcaklığı	0,00085	
	Deniz Suyu Tuzluluk Oranı	$8,0033 \times 10^{-7}$	
	Deniz Suyu Yoğunluğu	$2,36 \times 10^{-7}$	
	Deniz Trafığı Gürültüsü	$6,36 \times 10^{-12}$	
	Çevre Gürültüsü	$8,48 \times 10^{-10}$	
	Platformun Kendi Gürültüsü	$2,43 \times 10^{-13}$	
	Sismik Gürültü	0,029	
	Hedef Tonajı	$6,31 \times 10^{-6}$	
	Hedef Uzunluğu	$7,09 \times 10^{-6}$	
	T1	$4,21 \times 10^{-40}$	
	T2	$1,89 \times 10^{-16}$	
	T3	$5,21 \times 10^{-19}$	
Borderline SMOTE Yöntemiyle Üretilen Sentetik Veri	Sonar Menzili	1	Anlamlı Bir Fark Yoktur
	Sonar Yaşı	1	
	Sonar Frekansı	1	
	Deniz Suyu Sıcaklığı	1	
	Deniz Suyu Tuzluluk Oranı	1	
	Deniz Suyu Yoğunluğu	1	
	Deniz Trafığı Gürültüsü	1	
	Çevre Gürültüsü	1	
	Platformun Kendi Gürültüsü	1	
	Sismik Gürültü	1	
	Hedef Tonajı	1	
	Hedef Uzunluğu	1	
	T1	1	
	T2	1	
	T3	1	
ADASYN Yöntemiyle Üretilen Sentetik Veri	Sonar Menzili	1	Anlamlı Bir Fark Yoktur
	Sonar Yaşı	1	
	Sonar Frekansı	1	
	Deniz Suyu Sıcaklığı	1	
	Deniz Suyu Tuzluluk Oranı	1	
	Deniz Suyu Yoğunluğu	1	
	Deniz Trafığı Gürültüsü	1	
	Çevre Gürültüsü	1	
	Platformun Kendi Gürültüsü	1	
	Sismik Gürültü	1	
	Hedef Tonajı	1	
	Hedef Uzunluğu	1	

Tablo 7'nin Devamı

Veri Seti Adı	Sütun Adı	<i>p value</i> Değeri	Farklılık Durumu
ADASYN Yöntemiyle Üretilen Sentetik Veri	T1	$9,22 \times 10^{-52}$	Anlamlı Bir Fark Vardır
	T2	1	Anlamlı Bir Fark Yoktur
	T3	1	
ROSE Yöntemiyle Üretilen Sentetik Veri	Sonar Menzili	$1,98 \times 10^{-7}$	Anlamlı Bir Fark Vardır
	Sonar Yaşı	0,014	
	Sonar Frekansı	0,015	
	Deniz Suyu Sıcaklığı	$1,71 \times 10^{-6}$	
	Deniz Suyu Tuzluluk Oranı	$2,37 \times 10^{-5}$	
	Deniz Suyu Yoğunluğu	0,012	
	Deniz Trafiği Gürültüsü	0,002	
	Çevre Gürültüsü	0,011	
	Platformun Kendi Gürültüsü	$4,71 \times 10^{-5}$	
	Sismik Gürültü	0,0006	Anlamlı Bir Fark Yoktur
	Hedef Tonajı	0,082	
	Hedef Uzunluğu	0,082	
	T1	$1,19 \times 10^{-17}$	Anlamlı Bir Fark Vardır
	T2	0,000014	
	T3	0,00016	
Trim-SMOTE Yöntemiyle Üretilen Sentetik Veri	Sonar Menzili	$8,02 \times 10^{-133}$	Anlamlı Bir Fark Vardır
	Sonar Yaşı	$6,58 \times 10^{-98}$	
	Sonar Frekansı	$6,82 \times 10^{-37}$	
	Deniz Suyu Sıcaklığı	$1,53 \times 10^{-6}$	
	Deniz Suyu Tuzluluk Oranı	$5,24 \times 10^{-5}$	
	Deniz Suyu Yoğunluğu	$1,39 \times 10^{-12}$	
	Deniz Trafiği Gürültüsü	0,0001	
	Çevre Gürültüsü	$1,91 \times 10^{-10}$	
	Platformun Kendi Gürültüsü	$9,45 \times 10^{-22}$	
	Sismik Gürültü	$5,44 \times 10^{-7}$	
	Hedef Tonajı	$4,31 \times 10^{-14}$	
	Hedef Uzunluğu	$5,32 \times 10^{-10}$	
	T1	$3,15 \times 10^{-53}$	
	T2	$9,82 \times 10^{-44}$	
	T3	$2,42 \times 10^{-6}$	
MWMOTE Yöntemiyle Üretilen Sentetik Veri	Sonar Menzili	1	Anlamlı Bir Fark Yoktur
	Sonar Yaşı	1	
	Sonar Frekansı	1	
	Deniz Suyu Sıcaklığı	1	
	Deniz Suyu Tuzluluk Oranı	1	
	Deniz Suyu Yoğunluğu	1	
	Deniz Trafiği Gürültüsü	1	
	Çevre Gürültüsü	1	
	Platformun Kendi Gürültüsü	1	
	Sismik Gürültü	1	
	Hedef Tonajı	1	
	Hedef Uzunluğu	1	

Tablo 7'nin Devamı

Veri Seti Adı	Sütun Adı	<i>p value</i> Değeri	Farklılık Durumu
MWMOTE Yöntemiyle Üretilen Sentetik Veri	T1	1	Anlamlı Bir Fark Yoktur
	T2	1	
	T3	1	
HAC Yöntemiyle Üretilen Sentetik Veri	Sonar Menzili	0,864	Anlamlı Bir Fark Yoktur
	Sonar Yaşı	0,703	
	Sonar Frekansı	0,998	
	Deniz Suyu Sıcaklığı	0,581	
	Deniz Suyu Tuzluluk Oranı	0,897	
	Deniz Suyu Yoğunluğu	0,999	
	Deniz Trafiği Gürültüsü	0,938	
	Çevre Gürültüsü	0,999	
	Platformun Kendi Gürültüsü	0,531	
	Sismik Gürültü	0,848	
	Hedef Tonajı	0,226	
	Hedef Uzunluğu	0,226	
	T1	0,999	
	T2	0,864	
	T3	0,891	
MDO Yöntemiyle Üretilen Sentetik Veri	Sonar Menzili	1	Anlamlı Bir Fark Yoktur
	Sonar Yaşı	1	
	Sonar Frekansı	1	
	Deniz Suyu Sıcaklığı	1	
	Deniz Suyu Tuzluluk Oranı	1	
	Deniz Suyu Yoğunluğu	1	
	Deniz Trafiği Gürültüsü	1	
	Çevre Gürültüsü	1	
	Platformun Kendi Gürültüsü	1	
	Sismik Gürültü	1	
	Hedef Tonajı	1	
	Hedef Uzunluğu	1	
	T1	1	
	T2	1	
	T3	1	

Tablo 7’de yer alan sonuçlar incelendiğinde;

- Borderline SMOTE, MWMOTE, HAC ve MDO yöntemleriyle üretilen sentetik verilerin kaynak veri setindeki veriler arasında anlamlı bir fark bulunmamaktadır.

- SMOTE, ADASYN, ROSE ve Trim-SMOTE yöntemleriyle üretilen sentetik verilerin kaynak veri setindeki veriler arasında anlamlı bir fark bulunmaktadır.

Bu nedenle çalışmanın bundan sonraki bölümünde yapılacak testlerde sadece Borderline SMOTE, MWMOTE, HAC ve MDO yöntemiyle üretilmiş sentetik veriler dikkate alınmıştır.

2.2.2.4 Accuracy Skorlarının İncelenmesi

Accuracy (doğruluk) skoru, bir sınıflandırma modelinin ne kadar doğru çalıştığını ölçen bir performans metriğidir. Genellikle doğru tahminlerin toplam örnek sayısına olan oranını ifade eder. Bu skor, bir modelin tüm sınıflar üzerindeki başarı düzeyini tek bir sayıda özetler. Ancak, dengesiz sınıf dağılımlarına sahip veri setlerinde doğruluk skoru tek başına yeterli olmayabilir. Örneğin, eğer bir sınıf diğerlerine göre çok daha fazla örneğe sahipse, model bu sınıfa odaklanarak yüksek bir doğruluk elde edebilir, ancak diğer sınıfları düşük başarıyla sınıflandırabilir [49].

Bu çalışma kapsamında öncelikle yapay sinir ağları ile bir model eğitilmiş ve her bir veri seti için doğruluk skorları hesaplanmıştır. Elde edilen hesaplama sonuçları Tablo 8’de sunulmuştur.

Tablo 8: Doğruluk Skoru Değerleri.

Veri Seti Adı	Doğruluk Skoru
Borderline SMOTE Yöntemiyle Üretilen Sentetik Veri	0,805
MWMOTE Yöntemiyle Üretilen Sentetik Veri	0,803
HAC Yöntemiyle Üretilen Sentetik Veri	0,814
MDO Yöntemiyle Üretilen Sentetik Veri	0,871

2.2.2.5 F1 Skorlarının İncelenmesi

F1 skoru, bir sınıflandırma modelinin performansını ölçen bir istatistiksel metriktir ve genellikle dengesiz veri kümeleri üzerinde çalışan modellerin performansını değerlendirmek için kullanılır. F1 skoru, hassasiyet (precision) ve duyarlılık (recall) metriklerinin harmonik ortalaması olarak hesaplanır. Bu metrik, sınıflandırma problemlerinde doğru sonuçları değerlendirmek için tasarlanmıştır. Hassasiyet, doğru pozitif tahminlerin toplam pozitif tahminlere oranını temsil eder. Duyarlılık ise gerçek pozitif tahminlerin toplam gerçek pozitif durum sayısına oranını ifade eder. Hassasiyet ve duyarlılık arasında bir denge sağlayan F1 skoru, özellikle dengesiz sınıflandırma problemlerinde kullanıcıya daha kapsamlı bir değerlendirme sunar. Yüksek bir F1 skoru, modelin hem hassasiyeti hem de duyarlılığı göz önünde bulundurarak genel olarak iyi bir performansa işaret eder [49].

Bu çalışma kapsamında öncelikle yapay sinir ağları ile bir model eğitilmiş ve her bir veri seti için F1 skorları hesaplanmıştır. Elde edilen hesaplama sonuçları Tablo 9’da sunulmuştur.

Tablo 9: F1 Skoru Değerleri.

Veri Seti Adı	F1 Skoru
Borderline SMOTE Yöntemiyle Üretilen Sentetik Veri	0,891
MWMOTE Yöntemiyle Üretilen Sentetik Veri	0,890
HAC Yöntemiyle Üretilen Sentetik Veri	0,897
MDO Yöntemiyle Üretilen Sentetik Veri	0,896

2.2.2.6 Keskinlik Değerlerinin İncelenmesi

Keskinlik değeri istatistiksel bir ölçümdür ve genellikle sınıflandırma problemlerinde kullanılır. Özellikle makine öğrenimi ve veri madenciliği alanlarında, bir modelin performansını değerlendirmek için yaygın olarak kullanılır. Precision, modelin pozitif olarak sınıflandırdığı örneklerin ne kadarının gerçekten pozitif olduğunu ölçer. İdeal bir durumda, keskinlik değeri değeri 1'e yaklaşırken, kötü bir durumda 0'a yaklaşır. Keskinlik değeri, özellikle nadir olayların tespiti gereken durumlarda kullanışlıdır. Ancak, tek başına değerlendirildiğinde eksik kalabilir, bu nedenle genellikle diğer performans metrikleriyle birlikte değerlendirilir [49].

Bu çalışma kapsamında öncelikle yapay sinir ağları ile bir model eğitilmiş ve her bir veri seti için kesinlik değerleri hesaplanmıştır. Elde edilen hesaplama sonuçları Tablo 10'da sunulmuştur.

Tablo 10: Kesinlik Değerleri.

Veri Seti Adı	Kesinlik Değeri
Borderline SMOTE Yöntemiyle Üretilen Sentetik Veri	0,805
MWMOTE Yöntemiyle Üretilen Sentetik Veri	0,804
HAC Yöntemiyle Üretilen Sentetik Veri	0,814
MDO Yöntemiyle Üretilen Sentetik Veri	0,830

2.2.2.7 Duyarlılık Değerlerinin İncelenmesi

Bir sınıflandırma modelinin performansını değerlendiren bir istatistiksel ölçüdür. Bu ölçü, gerçek pozitif (true positive) sayısının toplam gerçek pozitif ve yanlış negatif (false negative) sayısına oranını ifade eder. Bu metrik, özellikle gerçek pozitif olguların kaçının doğru bir şekilde saptandığını ölçer. Yani, bir model ne kadar iyi "duyarlı" veya "hassas" bir şekilde pozitif durumları (bu çalışmada pozitif durumlar hedefin imha edilmesi olarak dikkate alınmıştır.) tespit edebiliyorsa, duyarlılık değeri o kadar yüksek olacaktır. Yüksek olması, modelin pozitif durumları kaçırmadığı ve bu durumları algılamada etkili olduğu anlamına gelir [49].

Bu çalışma kapsamında öncelikle yapay sinir ağları ile bir model eğitilmiş ve her bir veri seti için duyarlılık değerleri hesaplanmıştır. Elde edilen hesaplama sonuçları Tablo 11’de sunulmuştur.

Tablo 11: Duyarlılık Değerleri.

Veri Seti Adı	Duyarlılık Değeri
Borderline SMOTE Yöntemiyle Üretilen Sentetik Veri	1
MWMOTE Yöntemiyle Üretilen Sentetik Veri	0,996
HAC Yöntemiyle Üretilen Sentetik Veri	1
MDO Yöntemiyle Üretilen Sentetik Veri	0,996

2.2.2.8 ROC-AUC Değerlerinin İncelenmesi

Alıcı İşletim Karakteristiği (Receiver Operating Characteristic, ROC)-Eğrinin Altındaki Alan (Area Under the Curve , AUC), bir sınıflandırma modelinin performansını değerlendirmek için kullanılan bir ölçüdür. Bu ölçü, özellikle ikili sınıflandırma problemlerinde yaygın olarak kullanılır. ROC-AUC, bir modelin duyarlılık ve özgüllük performansını değerlendirir ve bu performansı tek bir ölçüde birleştirir. ROC, bir sınıflandırma modelinin farklı kesme noktalarında duyarlılık ve özgüllük arasındaki ilişkiyi gösteren bir grafikdir. Kesme noktası, modelin pozitif veya negatif olarak sınıflandırma eşiğini belirtir. ROC eğrisi, bu kesme noktalarını temsil eden bir çizgidir. Eğri, sol üst köşeden sağ alt köşeye doğru ilerledikçe modelin performansının arttığını gösterir. ROC-AUC, ROC eğrisinin altındaki alanı ifade eder. Bu değer, modelin tüm duyarlılık ve özgüllük seviyelerindeki performansını birleştirir. ROC-AUC değeri 0 ile 1 arasında değişir, 1'e ne kadar yakınsa model o kadar iyi performans gösterir [51].

Bu çalışma kapsamında öncelikle yapay sinir ağları ile bir model eğitilmiş ve her bir veri seti için ROC-AUC değerleri hesaplanmıştır. Elde edilen hesaplama sonuçları Tablo 12’de sunulmuştur.

Tablo 12: ROC-AUC Değerleri.

Veri Seti Adı	ROC-AUC Değeri
Borderline SMOTE Yöntemiyle Üretilen Sentetik Veri	0,533
MWMOTE Yöntemiyle Üretilen Sentetik Veri	0,512
HAC Yöntemiyle Üretilen Sentetik Veri	0,487
MDO Yöntemiyle Üretilen Sentetik Veri	0,481

Buraya kadar açıklanan tüm hesaplamalar göz önünde bulundurulduğunda;

- Histogram-Yoğunluk Grafiklerine göre kaynak veri setine en yakın verilerin MDO yöntemiyle üretildiği,
- Q-Q çizgilerine göre kaynak veri setine en yakın verilerin MDO yöntemiyle üretildiği,
- Accuracy skorlarına göre kaynak veri setine en yakın verilerin MDO yöntemiyle üretildiği,
- F1 skorlarına göre kaynak veri setine en yakın verilerin HAC yöntemiyle üretildiği,
- Kesinlik değerlerine göre kaynak veri setine en yakın verilerin MDO yöntemiyle üretildiği,
- Duyarlılık değerlerine göre kaynak veri setine en yakın verilerin Borderline SMOTE ve HAC yöntemleriyle üretildiği,
- ROC-AUC değerlerine göre kaynak veri setine en yakın verilerin Borderline SMOTE yöntemiyle üretildiği görülmektedir. Bu nedenle çalışmada MDO yöntemiyle üretilen sentetik veriler kullanılmıştır.

BÖLÜM III

ATILACAK MÜHİMMAT SAYILARI VE TEHDİT DENİZALTI TESPİT OLASILIKLARININ HESAPLANMASI

3.1 ATILACAK TORPİDO SAYILARININ HESAPLANMASI

Çalışmada kullanılacak verilerin oluşturulmasını müteakip her bir tehdit denizaltının imhası için atılması gereken mühimmat sayıları ayrı ayrı hesaplanmıştır. Söz konusu hesaplamalarda makine öğrenme algoritmalarından faydalanılmıştır. Çalışmada kullanılan algoritmalara ilişkin detaylar müteakip maddelerde açıklanmıştır.

3.1.1 Makine Öğrenmesinin İncelenmesi

Makine öğrenmesi, bilgisayar sistemlerinin örüntüleri ve bilgileri analiz ederek deneyimlerden öğrenme yeteneği kazandığı bir yapay zeka dalıdır. Bu alandaki temel amaç, bir programın belirli bir görevi daha iyi yapabilmesi için veri üzerinden öğrenme sağlamaktır. Geleneksel programlamadan farklı olarak, makine öğrenmesi algoritmaları belirli bir görev için doğrudan programlanmaz; bunun yerine, veri üzerinden öğrenerek kendilerini geliştirirler [51, 52].

Makine öğrenmesi, geniş veri setlerini analiz edebilme yeteneği sayesinde karmaşık modellerin oluşturulmasını mümkün kılar. Bu veri setleri, algoritmaların öğrenmesi için kullanılır ve daha sonra yeni verilerle karşılaşıldığında bu modellerden yararlanılarak çeşitli görevler gerçekleştirilir. Sınıflandırma, regresyon, kümeleme ve öneri sistemleri gibi birçok farklı görev makine öğrenmesi uygulamalarının temelini oluşturur [51, 52].

Makine öğrenmesinin bir alt dalı olan derin öğrenme, yapay sinir ağlarını kullanarak daha karmaşık ve katmanlı modeller oluşturmayı amaçlar. Derin öğrenme algoritmaları, büyük veri setlerinden öğrenme ve karmaşık ilişkileri anlama konusunda başarı elde etmiştir. Yapay zekâ ve nesne tanıma gibi alanlarda derin öğrenme önemli başarılar elde etmiştir [51, 52].

Makine öğrenimi uygulamaları birçok sektörde kullanılmaktadır. Finans, sağlık, perakende, ulaşım ve enerji gibi birçok endüstri, veri analizi ve öğrenme algoritmalarını kullanarak iş süreçlerini iyileştirmekte ve karar verme süreçlerini optimize etmektedir. Örneğin, müşteri tercihlerini anlamak, talep tahminleri yapmak ve otomatik sürüş teknolojileri gibi birçok alanda makine öğrenimi büyük bir etki yaratmıştır [51, 52].

Ancak, makine öğrenimi uygulamalarının başarılı olabilmesi için kaliteli verilere ihtiyaç vardır. Veri setlerinin büyüklüğü ve kalitesi algoritmaların başarısını etkileyen kritik faktörlerdir. Ayrıca, etik ve gizlilik gibi konular da makine öğrenimi uygulamalarının geliştirilmesi sırasında dikkate alınması gereken önemli sorunlardır [51, 52].

Makine öğrenimi, teknolojik gelişmelerle birlikte sürekli olarak evrim geçiren bir alan olup, bu alandaki uzmanların ve araştırmacıların yeni yöntemler ve algoritmalar geliştirmek için çalışmalarını sürdürmeleri beklenmektedir. Bu sayede, gelecekte daha karmaşık problemlere çözüm getirebilen ve insanlarla etkileşimde bulunabilen daha akıllı makineler ve sistemler ortaya çıkabilir [51, 52].

3.1.2 Çalışmada Kullanılan Makine Öğrenme Algoritmalarının İncelenmesi

Çalışmada kapsamında;

- K En Yakın Komşu (K-Nearest Neighbor, KNN),
- Karar Ağaçları (Decision Tree),
- Rastgele Orman (Random Forest),
- Gradyan Artırma (Gradient Boosting),
- Yapay Sinir Ağları (Artificial Neural Networks, ANN),
- Yukarıda yer algoritmalarından müteşekkil Stacking yöntemini kullanan

karışım (mix) algoritma,

- Yukarıda yer algoritmalarından müteşekkil Voting yöntemini kullanan

karışım (mix) algoritmalarından faydalanılmıştır.

3.1.2.1 KNN Algoritması

Bu algoritma, bir veri noktasının tahmin edilmesi için en yakın komşularının etkisini kullanır. Temel amacı, bir veri noktasının çıktısını, çevresindeki K en yakın komşusunun çıktılarının birleşimiyle tahmin etmektir [53, 54].

K-NN regresyonun avantajları şunlardır:

- Basit ve anlaşılır bir algoritmadır.
- Model eğitimi aşamasında karmaşık bir optimizasyon süreci yoktur.
- Non-lineer ilişkileri yakalamada iyi performans gösterebilir.

Ancak, dezavantajları da vardır:

- Veri seti büyüdükçe hesaplama maliyeti artar.
- Eğitim verileri büyükse ve boyutluysa, hafıza sorunları ortaya çıkabilir.
- Optimal K değerini seçmek zordur.

K-NN regresyonu genellikle küçük veri setleri ve basit regresyon problemleri için tercih edilir. Optimal K değerinin seçimi, genellikle çapraz doğrulama gibi yöntemlerle belirlenir [53, 54].

3.1.2.2 Karar Ağaçları Algoritması

Karar ağaçları, hem sınıflandırma hem de regresyon problemlerini çözmek için kullanılan bir makine öğrenimi algoritmasıdır. Bu algoritma, veri setini öğrenme sürecinde ağaç yapısı şeklinde temsil eder. Her bir iç düğüm (node) bir özellik (feature) testini temsil eder, her bir kenar (edge) bir test sonucunu ve her bir yaprak düğüm (leaf node) de bir tahmini değeri içerir. Regresyon problemlerinde, karar ağaçları, sürekli bir çıktı değişkenini tahmin etmek için kullanılır. Örneğin, bir evin satış fiyatını tahmin etmek için kullanılabilir. Bu durumda, ağaç, özelliklerin (örneğin, evin büyüklüğü, oda sayısı, konumu vb.) bir dizi testine dayanarak evin tahmini satış fiyatını belirler. Karar ağacı algoritması genellikle aşağıdaki adımları içerir:

- Kök düğüm oluşturma: Veri seti üzerinde en iyi bölünmeyi seçerek başlar. En iyi bölünme, veriyi mümkün olduğunca homojen alt gruplara ayıran bir özellik ve eşik değerini içerir.
- Alt düğümlerin oluşturulması: Kök düğümdeki bölünme işlemi, alt düğümleri (iç düğümler veya yaprak düğümler) oluşturur. Bu adım, her bir alt düğümde aynı işlemi tekrarlar.
- Ağacın büyütülmesi: Belirli bir durma kriterine ulaşıncaya veya bölünmeler artık veriyi daha homojen hale getiremeyecek hale gelene kadar ağacın büyümesi devam eder.

- Ağacın kırılması (pruning): Ağacın aşırı uyum (overfitting) problemini önlemek için gerektiğinde kırılabilir. Bu, gereksiz düğümlerin veya dalların kaldırılmasını içerir [55, 56].

Karar ağaçları, yorumlana bilirlilikleri, kullanım kolaylıkları ve veriye uyum yetenekleri nedeniyle popülerdir. Ancak, tek başına kullanıldıklarında bazen aşırı uyuma eğilimli olabilirler. Bu nedenle, genellikle birçok ağacın bir araya getirilerek oluşturulan bir orman yapısı olan "rastgele ormanlar" gibi yöntemlerle birlikte kullanılırlar [55, 56].

3.1.2.3 Rastgele Orman Algoritması

Rastgele Orman, bir dizi karar ağacının bir araya getirilmesiyle oluşturulan bir modeldir. Bu yöntem, özellikle karmaşık ve yüksek boyutlu veri setlerinde etkili olan bir makine öğrenimi tekniğidir [57, 58]. Rastgele Orman algoritmasının ana özellikleri:

- Temelde bir dizi karar ağacının bir araya getirilmesiyle oluşur. Her bir karar ağacı, veri setini belirli özelliklere göre böler ve her bir bölüm için bir tahmin yapar.
- Veri setini sınıflara ayırmak veya regresyon için tahminlerde bulunmak için kullanılır. Her bir düğüm, bir özelliği değerlendirir ve bir karar alır, bu sayede veriyi daha küçük alt gruplara böler.
- "Rastgele" kelimesi, bu yöntemin temel özelliğini ifade eder. Rastgele Orman, farklı alt veri setlerini ve özellik alt kümelerini kullanarak bir dizi karar ağacı oluşturur. Bu rastgelelik, genellikle özellik seçimindeki rastgelelikle birleşir. Her bir karar düğümünde, bir alt küme özellik rastgele seçilir ve bu özelliklere göre veri seti bölünür.
- Rastgele Orman, bagging tekniği kullanır. Bu, aynı veri seti üzerinde farklı alt veri setleri oluşturularak ve her alt veri setini kullanarak ayrı ayrı karar ağaçları eğiterek modelin genelleştirilebilirliğini artırır. Her alt veri seti, rastgele seçilen örneklerin bir alt kümesini içerir. Bu işlem, örneklerin tekrar izin verilen bir yöntem olan "bootstrap" kullanılarak gerçekleştirilir.
- Her bir karar ağacının tahminlerini birleştirerek final tahminleri yapar. Sınıflandırma problemlerinde, sıklıkla oylama yöntemi kullanılır; regresyon problemlerinde ise genellikle tahminlerin ortalaması alınır.

- Birçok ağacın bir araya getirilmesiyle oluşturulduğu için tek bir ağaca göre daha az aşırı uyma eğilimindedir. Bu, modelin daha genelleştirilebilir olmasına yardımcı olabilir.
- Özellikle büyük, karmaşık veri setleri üzerinde iyi performans gösteren bir modeldir. Rastgele seçim ve bagging kullanımı, modelin dayanıklılığını artırarak genel performansını iyileştirir [57, 58].

3.1.2.4 Gradient Boosting Algoritması

Gradient Boosting, makine öğrenimi alanında kullanılan güçlü bir toplu öğrenme yöntemidir. Temel amacı, zayıf öğrencileri (genellikle zayıf karar ağaçları olarak adlandırılır) bir araya getirerek güçlü bir tahmin modeli oluşturmaktır. Gradient Boosting algoritmaları, özellikle regresyon ve sınıflandırma problemleri için başarılı bir şekilde kullanılmaktadır. Gradient Boosting'in temel fikri, her ağacın önceki ağaçların hatasını düzeltmeye odaklanmasıdır [59, 60]. Gradient Boosting'in çalışma prensibi:

- İlk Ağaç Oluşturma (Base Learner): İlk ağaç, veri setinin üzerinde eğitilir ve bir tahmin yapar. Ancak, bu tahmin genellikle hatalıdır.
- Hata Hesaplama (Loss Calculation): İlk ağacın tahminleri ile gerçek değerler arasındaki hatalar hesaplanır. Hatalar, modelin performansını ölçmek için kullanılır.
- Hata Hesaplanan Ağaç (Weighted Tree): İkinci ağaç, hataları düzeltmeye odaklanarak inşa edilir. İlk ağacın tahminlerine olan hataların ağırlıklarını kullanarak bu hataları düzeltmeye çalışır.
- Toplam Hata Hesaplanması (Total Loss Calculation): İlk iki ağacın tahminlerini toplayarak yeni bir tahmin yapılır. Bu yeni tahminle gerçek değerler arasındaki hatalar hesaplanır.
- Yeni Ağaçların Eklenmesi (Iterative Process): Bu süreç belirlenen bir sayıda veya belirli bir hata eşiğine ulaşılan kadar tekrarlanır. Her aşamada, bir sonraki ağaç önceki ağaçların hatalarını düzeltmeye çalışır.

Gradient Boosting algoritmaları, genellikle öğrenme hızı (learning rate) gibi hiperparametreleri ayarlamak ve ağaç sayısını kontrol etmek gibi çeşitli hiperparametre ayarlarına sahiptir. XGBoost, LightGBM, ve CatBoost gibi popüler kütüphaneler, Gradient Boosting'in optimize edilmiş versiyonlarını içerir ve

genellikle büyük veri setlerinde ve rekabetçi makine öğrenimi problemlerinde tercih edilir [59, 60].

3.1.2.5 ANN

Biyolojik sinir ağlarından esinlenerek oluşturulmuş bir makine öğrenimi modelidir. Bu model, karmaşık görevleri gerçekleştirmek veya desenleri tanımak gibi birçok uygulama için kullanılır. Yapay Sinir Ağları, bilgisayarlar arasında bilgi işleme ve öğrenme kapasitesini simüle etmek amacıyla geliştirilmiştir. Yapay Sinir Ağları, birçok nöron adı verilen basit birimlerden oluşan katmanlar halinde organize edilmiştir. Bu nöronlar, girdileri alır, bunları işler ve ardından bir çıkış üretirler. Bu yapının temel amacı, karmaşık ilişkileri ve desenleri öğrenme yeteneği sunmaktır [61, 62, 63].

ANN'nin temel bileşenleri:

- Giriş Katmanı (Input Layer): Dış dünyadan gelen verilerin sisteme girdiği katmandır. Her bir girdi bir nörona bağlıdır.
- Gizli Katmanlar (Hidden Layers): Giriş katmanından alınan veriler, bir veya birden çok gizli katman boyunca işlenir. Her bir gizli katman, karmaşıklığı artırarak ve daha yüksek düzeyde özellikleri çıkararak veriyi işler.
- Çıkış Katmanı (Output Layer): Ağın son katmanıdır ve sonuçları üretir. Bu katmandaki nöronlar, genellikle belirli bir görevle ilgili çıkışları temsil eder.
- Ağırlıklar (Weights) ve Biaslar (Biases): Her nöron arasındaki bağlantılar ağırlıklarla temsil edilir. Ağırlıklar, girdileri çıkışa dönüştürmek için kullanılır. Biaslar, nöronların aktivasyon eşiğini ayarlar.
- Aktivasyon Fonksiyonları: Her nöron, girdileri ve ağırlıkları kullanarak bir çıkış üretir. Bu çıkış, genellikle bir aktivasyon fonksiyonu tarafından düzenlenir. Sigmoid, ReLU, tanh gibi fonksiyonlar yaygın olarak kullanılır.
- Eğitim Algoritması: Ağı eğitmek için kullanılan algoritmalardır. Geriye Yayılım (Backpropagation) en yaygın kullanılan eğitim algoritmasıdır. Bu algoritma, ağı bir dizi öğrenme örneği üzerinden eğitir ve hata oranını azaltmaya çalışır.

Yapay Sinir Ağları, görüntü ve ses tanıma, doğal dil işleme, oyun stratejileri oluşturma, tahmin yapma gibi birçok uygulama alanında başarıyla kullanılmaktadır.

Ancak, aşırı öğrenme, eğitim verisi eksikliği gibi sorunlarla başa çıkma ihtiyacı vardır ve bu nedenle ANN'yi kullanırken dikkatli olunmalıdır [61, 62, 63].

3.1.2.6 Stacking Yöntemiyle Oluşturulmuş Mix Algoritma

Stacking, birçok makine öğrenimi modelini bir araya getirerek daha güçlü ve geliştirilebilir bir model oluşturmayı amaçlayan bir ensemble öğrenme yöntemidir. Stacking, temel modellerin (base models) tahminlerini kullanarak bir meta-model öğrenir. Bu meta-model, temel modellerin çıkışlarına dayalı olarak daha yüksek düzeyde öğrenme yapabilir ve genel performansı artırabilir [64]. Stacking'in genel adımları şunlardır:

- **Veri Setini Bölme:** Veri seti genellikle eğitim ve test setlerine bölünür. Eğitim seti, temel modellerin eğitildiği veri kümesidir. Test seti, meta-modelin performansının değerlendirildiği veri kümesidir.
- **Temel Modellerin Eğitimi:** İlk aşamada, bir dizi temel model eğitilir. Bu temel modeller genellikle farklı algoritmalar veya farklı hiperparametre ayarları kullanılarak oluşturulur. Temel modeller, eğitim seti üzerinde tahminler yapar.
- **Temel Modellerin Tahminleri:** Eğitilen temel modeller, test seti üzerinde tahminlerde bulunur.
- **Meta-Modelin Eğitimi:** Temel modellerin tahminleri, meta-model tarafından kullanılmak üzere bir araya getirilir. Bu tahminler, eğitim seti üzerindeki gerçek hedef değerleri ile kullanılarak meta-model eğitilir.
- **Meta-Modelin Değerlendirilmesi:** Meta-modelin performansı test seti üzerinde değerlendirilir.
- **Stacking'in Uygulanması:** Gerçek uygulamada, yeni veri noktaları üzerinde tahmin yapmak için önce temel modellerin ardından meta-model kullanılır.

Stacking'in avantajları şunlardır:

- Daha yüksek tahmin performansı ve genelleme yeteneği.
- Farklı algoritmaların birleştirilmesi, bir modelin diğerinin zayıf yönlerini telafi etmesine olanak tanır.
- Overfitting'e karşı daha dirençli olabilir.

Ancak, bu avantajlarla birlikte dikkate alınması gereken bazı hususlar vardır:

- Daha karmaşık bir model yaratma potansiyeli, eğitim süresini artırabilir.

- Modelin karmaşıklığı arttıkça, yeterli veri olmadığında aşırı uydurma riski artabilir.

- Veri seti üzerindeki temel modellerin çeşitliliği, genellikle stacking'in başarısını artırabilir [64].

Bu kapsamda çalışmada KNN, karar ağaçları, rastgele orman, gradient boosting ve ANN algoritmalarından müteşekkil ve ANN regresyon modelinin meta-model olduğu bir Stacking algoritma tasarlanmıştır.

3.1.2.7 Voting Yöntemiyle Oluşturulmuş Mix Algoritma

Genellikle makine öğrenimi ve veri madenciliği alanlarında kullanılan bir kavramdır. Bu terim, bir dizi farklı öğrenme algoritmasının bir araya getirilerek ortak bir karar algoritması oluşturulmasını ifade eder. Bu, bir tür ensemble öğrenme yöntemidir. Voting algoritmaları, genellikle bir dizi alt modelin tahminlerini bir araya getirerek daha güvenilir ve genelde daha doğru bir tahmin elde etmeyi amaçlar. Bu algoritmalar, çeşitli öğrenme algoritmalarını veya aynı algoritmanın farklı parametrelerle eğitilmiş versiyonlarını içerebilir. İki temel türde voting yöntemi bulunmaktadır [64]:

- Hard Voting (Sert Oy): Her bir alt model kendi tahminini yapar, ardından çoğunluk oyunu olarak final tahmini belirler. Örneğin, bir sınıflandırma problemi düşünün; her bir alt model, bir veri noktasını belirli bir sınıfa tahmin eder ve ardından en çok oy alan sınıf final tahmin olarak seçilir [64].

- Soft Voting (Yumuşak Oy): Her bir alt model, her bir sınıf için tahmin güvenliğini içeren bir olasılık dağılımı üretir. Bu olasılıkların ortalaması alınarak en yüksek olasılığa sahip sınıf final tahmin olarak seçilir. Soft voting, özellikle alt modellerin tahmin güvenilirliği hakkında bilgi sağladığında kullanışlıdır [64].

Mix algoritma, genellikle farklı türde öğrenme algoritmalarını (örneğin, karar ağaçları, destek vektör makineleri, KNN, vb.) veya aynı algoritmanın farklı hiperparametre ayarlarıyla eğitilmiş modelleri bir araya getirerek daha sağlam ve genelleştirilebilir bir model elde etmeyi amaçlar [64].

Bu tür ensemble (ensamble) yöntemleri, tek bir modelin performansını aşmaya çalışır ve genellikle daha güçlü, daha istikrarlı ve daha genel modeller elde etmede başarılı olabilir. Bu yaklaşım, "bilge kalabalıkların" etkisini taklit eder, yani bir grup insanın genellikle bireylerden daha iyi kararlar alabilmesi gibi, bir grup modelin de bireylerden daha iyi tahminlerde bulunabilmesini amaçlar [64].

Bu kapsamda çalışmada KNN, karar ağaçları, rastgele orman, gradient boosting ve ANN algoritmalarından müteşekkil hard voting algoritması tasarlanarak tahminlemeler yapılmıştır.

3.1.3 Algoritma Performanslarının Karşılaştırılması ve En İyi Performans Gösteren Algoritmanın Belirlenmesi

Çalışma kapsamında yukarıda detayları açıklanan algoritmalar ve MDO yöntemiyle oluşturulan sentetik veriler kullanılarak tahminler yapılmıştır. Ardından hangi modelin daha iyi performans gösterdiğinin belirlenebilmesi için iki farklı test yapılmıştır. İlk testte modelin eğitildiği veri seti Ayırarak Çapraz Doğrulama (Holdout Cross Validation)¹ yöntemiyle %80 eğitim %20 test olacak şekilde ayrılmıştır. Verilerin %80'i ile eğitilen algoritmalar kalan %20'lik veriler ile test edilmiştir. Bu testler sonucunda tahminlerin doğrulukları Hata Kareleri Ortalaması (Mean Squared Error, MSE)² skorlarına göre belirlenmiştir [65]. İkinci testte ise dost kuvvet, düşman kuvvet ve çevresel verilerden oluşturulan ve Tablo 14'te sunulan "Tahmin Veri Seti"ndeki değerler kullanılarak yeniden tahminler yapılmıştır. Yapılan tahminlerin doğrulukları yine MSE skorlarına bakılarak hesaplanmıştır. Bu sayede modelin veri değişimindeki performansı da gözlemlenebilmiştir. Söz konusu hesaplamalara ilişkin sonuçlar Tablo 13'te sunulmuştur.

Tablo 13: Makine Öğrenme Algoritmaları Tahmin Doğruluk Oranları.

Eğitim ve Test Verileri Aynı Veri Setinden Alınan Durum		Eğitim ve Test Verileri Farklı Veri Setinden Alınan Durum	
Algoritmanın Adı	MSE Skoru	Algoritmanın Adı	MSE Skoru
KNN	4,23	KNN	4,02
Karar Ağaçları	4,24	Karar Ağaçları	4,03
Rastgele Orman	4,24	Rastgele Orman	3,07
Gradient Boasting	3,65	Gradient Boasting	3,24
Yapay Sinir Ağları	2,81	Yapay Sinir Ağları	4,63
Stacking	4,02	Stacking	4,02
Voting	3,67	Voting	3,02

¹ Ayırarak Çapraz Doğrulama (Holdout Cross Validation) yöntemi büyük veri setlerinde etkin bir yöntem olduğundan bizim çalışmamız için doğru sonuçlar vereceği değerlendirilmiş ve hesaplama dahil edilmiştir.

² Ortalama Kare Hatası tahmin edilen sonuçlarınızın gerçek sayıdan ne kadar farklı olduğuna dair size mutlak bir sayı verir. Tek bir sonuçtan çok fazla içgörü yorumlayamazsınız, ancak size diğer model sonuçlarıyla karşılaştırmak için gerçek bir sayı verir ve en iyi regresyon modelini seçmenize yardımcı olur [65].

Tablo 13'teki sonuçlar incelendiğinde;

- Birinci testte en iyi sonucun yapay sinir ağları algoritması ile elde edildiği,
- İkinci testte ise en iyi sonucun voting yöntemiyle oluşturulmuş mix algoritma ile elde edildiği görülmektedir.

Bu kapsamda tüm algoritmaların her iki testte gösterdikleri performans incelendiğinde birinci testte en iyi sonucu veren yapay sinir ağlarının ikinci testte çok fazla hata yaptığı görülmektedir. Buna karşın voting yöntemiyle oluşturulmuş mix algoritmanın ise her iki testte de hata oranı daha azdır. Bu nedenle atılması gereken mühimmat sayılarının hesaplanmasında anılan algorithmadan faydalanılmıştır.



Tablo 14: Dost Kuvvet Tahmin Veri Seti Örnek Gösterimi.

Sonar Menzili	Sonar Yaşı	Sonar Frekansı	Deniz Suyu Sıcaklığı	Deniz Suyu Tuzluluk Oranı	Deniz Suyu Yoğunluğu	Deniz Trafikği Gürültüsü	Çevresel Gürültüler	Platformun Kendi Gürültüsü	Sismik Gürültüler	Tehdit Denizaltı Tonajı	Tehdit Denizaltı Uzunluğu
46	45	6.7	15	32	29.3	66.16	35	30	1,2	1100	60
46	45	6.7	15	32	29.3	66.16	35	30	1,2	1100	60
46	45	6.7	15	32	29.3	66.16	35	30	1,2	1100	60
46	45	6.7	15	32	29.3	66.16	35	30	1,2	1100	60
46	45	6.7	15	32	29.3	66.16	35	30	1,2	1800	75
46	45	6.7	15	32	29.3	66.16	35	30	1,2	1800	75
67	37	8.2	15	32	29.3	66.16	35	30	1,2	1800	75
67	37	8.2	15	32	29.3	66.16	35	30	1,2	1800	75
67	37	8.2	15	32	29.3	66.16	35	30	1,2	1800	75
48	24	7	15	32	29.3	66.16	35	30	1,2	1200	60
48	24	7	15	32	29.3	66.16	35	30	1,2	1200	60
48	24	7	15	32	29.3	66.16	35	30	1,2	1200	60
44	8	7	15	32	29.3	66.16	35	30	1,2	1200	60
44	8	7	15	32	29.3	66.16	35	30	1,2	1200	60
44	8	7	15	32	29.3	66.16	35	30	1,2	1500	70
44	8	7	15	32	29.3	66.16	35	30	1,2	1500	70
56	35	8	15	32	29.3	66.16	35	30	1,2	1500	70
56	35	8	15	32	29.3	66.16	35	30	1,2	1500	70
56	35	8	15	32	29.3	66.16	35	30	1,2	1400	65
56	35	8	15	32	29.3	66.16	35	30	1,2	1400	65
56	35	8	15	32	29.3	66.16	35	30	1,2	1400	65

3.1.4 Hedefin İmhası İçin Atılması Gerekli Mühimmat Sayılarının Hesaplanması

Çalışmada kullanılacak algoritmanın belirlenmesinin ardından her bir tehdit hedefin imha edilebilmesi için atılması gereken mühimmat sayıları hesaplanmıştır. Bahse konu hesaplama sonuçları Tablo (15 ve 16)'da sunulmuştur.

Tablo 15: Dost Kuvvet Tarafından Atılması Gerekli Mühimmat Sayıları.

Atıcı Platform	Hedef Platform	T1	T2	T3
Fırkateyn Tip1	D/A Tip1	1	2	0
Fırkateyn Tip1	D/A Tip2	2	2	0
Fırkateyn Tip1	D/A Tip3	1	2	0
Fırkateyn Tip1	D/A Tip4	2	2	0
Fırkateyn Tip1	D/A Tip5	1	2	0
Fırkateyn Tip2	D/A Tip1	0	2	1
Fırkateyn Tip2	D/A Tip2	0	2	1
Fırkateyn Tip2	D/A Tip3	0	2	1
Fırkateyn Tip2	D/A Tip4	0	2	2
Fırkateyn Tip2	D/A Tip5	0	2	1
Fırkateyn Tip3	D/A Tip1	1	0	3
Fırkateyn Tip3	D/A Tip2	2	0	1
Fırkateyn Tip3	D/A Tip3	2	0	1
Fırkateyn Tip3	D/A Tip4	2	0	3
Fırkateyn Tip3	D/A Tip5	1	0	1
Korvet	D/A Tip1	0	2	1
Korvet	D/A Tip2	0	2	2
Korvet	D/A Tip3	0	3	1
Korvet	D/A Tip4	0	2	1
Korvet	D/A Tip5	0	2	1
Helikopter Tip1	D/A Tip1	0	3	2
Helikopter Tip1	D/A Tip2	0	2	1
Helikopter Tip1	D/A Tip3	0	2	1
Helikopter Tip1	D/A Tip4	0	2	1
Helikopter Tip1	D/A Tip5	0	2	2
Helikopter Tip2	D/A Tip1	2	1	0
Helikopter Tip2	D/A Tip2	2	2	0
Helikopter Tip2	D/A Tip3	2	1	0
Helikopter Tip2	D/A Tip4	2	2	0
Helikopter Tip2	D/A Tip5	2	1	0
Deniz Karakol Uçağı	D/A Tip1	1	0	3
Deniz Karakol Uçağı	D/A Tip2	2	0	1
Deniz Karakol Uçağı	D/A Tip3	2	0	1
Deniz Karakol Uçağı	D/A Tip4	2	0	3
Deniz Karakol Uçağı	D/A Tip5	1	0	1
Denizaltı Tip1	D/A Tip1	0	3	2
Denizaltı Tip1	D/A Tip2	0	2	1
Denizaltı Tip1	D/A Tip3	0	2	1
Denizaltı Tip1	D/A Tip4	0	2	1

Tablo 15'in Devamı

Atıcı Platform	Hedef Platform	T1	T2	T3
Denizaltı Tip2	D/A Tip5	1	2	0
Denizaltı Tip3	D/A Tip1	2	0	0
Denizaltı Tip3	D/A Tip2	2	0	0
Denizaltı Tip3	D/A Tip3	2	0	0
Denizaltı Tip3	D/A Tip4	2	0	0
Denizaltı Tip3	D/A Tip5	2	0	1
Denizaltı Tip4	D/A Tip1	2	2	0
Denizaltı Tip4	D/A Tip2	2	1	0
Denizaltı Tip4	D/A Tip3	3	2	0
Denizaltı Tip4	D/A Tip4	2	2	0
Denizaltı Tip4	D/A Tip5	2	1	0

Tablo 16: Düşman Kuvvet Tarafından Atılması Gereken Mühimmat Sayıları.

Atıcı Platform	Hedef Platform	T4	T5	T6
D/A Tip1	Fırkateyn Tip1	4	1	0
D/A Tip1	Fırkateyn Tip2	4	1	0
D/A Tip1	Fırkateyn Tip3	3	1	0
D/A Tip1	Korvet	3	1	0
D/A Tip1	Helikopter Tip1	4	1	0
D/A Tip1	Helikopter Tip2	4	1	0
D/A Tip1	Deniz Karakol Uçağı	3	0	0
D/A Tip1	Denizaltı Tip1	3	1	0
D/A Tip1	Denizaltı Tip2	4	1	0
D/A Tip1	Denizaltı Tip3	3	1	0
D/A Tip1	Denizaltı Tip4	4	0	0
D/A Tip2	Fırkateyn Tip1	0	4	1
D/A Tip2	Fırkateyn Tip2	0	2	1
D/A Tip2	Fırkateyn Tip3	0	3	1
D/A Tip2	Korvet	0	3	1
D/A Tip2	Helikopter Tip1	0	2	3
D/A Tip2	Helikopter Tip2	0	2	3
D/A Tip2	Deniz Karakol Uçağı	0	2	3
D/A Tip2	Denizaltı Tip1	0	3	1
D/A Tip2	Denizaltı Tip2	0	2	1
D/A Tip2	Denizaltı Tip3	0	2	1
D/A Tip2	Denizaltı Tip4	0	2	1
D/A Tip3	Fırkateyn Tip1	3	0	1
D/A Tip3	Fırkateyn Tip2	3	0	1
D/A Tip3	Fırkateyn Tip3	3	0	1
D/A Tip3	Korvet	4	0	1
D/A Tip3	Helikopter Tip1	3	0	1
D/A Tip3	Helikopter Tip2	4	0	2
D/A Tip3	Deniz Karakol Uçağı	4	0	2
D/A Tip3	Denizaltı Tip1	4	0	2
D/A Tip3	Denizaltı Tip2	2	0	3
D/A Tip3	Denizaltı Tip3	3	0	1
D/A Tip3	Denizaltı Tip4	4	0	1

Tablo 16'nın Devamı

Atıcı Platform	Hedef Platform	T4	T5	T6
D/A Tip4	Fırkateyn Tip1	3	0	1
D/A Tip4	Fırkateyn Tip2	2	4	0
D/A Tip4	Fırkateyn Tip3	2	4	0
D/A Tip4	Korvet	3	2	0
D/A Tip4	Helikopter Tip1	2	3	0
D/A Tip4	Helikopter Tip2	3	2	0
D/A Tip4	Deniz Karakol Uçağı	3	2	0
D/A Tip4	Denizaltı Tip1	3	2	0
D/A Tip4	Denizaltı Tip2	1	2	0
D/A Tip4	Denizaltı Tip3	3	1	0
D/A Tip4	Denizaltı Tip4	2	2	0
D/A Tip5	Fırkateyn Tip1	2	2	0
D/A Tip5	Fırkateyn Tip2	0	4	2
D/A Tip5	Fırkateyn Tip3	0	4	2
D/A Tip5	Korvet	0	2	2
D/A Tip5	Helikopter Tip1	0	3	2
D/A Tip5	Helikopter Tip2	0	2	2
D/A Tip5	Deniz Karakol Uçağı	0	2	2
D/A Tip5	Denizaltı Tip1	0	2	2
D/A Tip5	Denizaltı Tip2	0	2	3
D/A Tip5	Denizaltı Tip3	0	2	1
D/A Tip5	Denizaltı Tip4	0	3	1

3.2 TEHDİT DENİZALTI TESPİT OLASILIKLARININ HESAPLANMASI

Çalışma kapsamında pekiştirmeli öğrenme algoritmasına girdi oluşturacak bir diğer parametre de tehdit denizaltı tespit olasılıklarıdır. Tespit edilme olasılığı daha düşük olan bir denizaltı daha iyi bir teknolojiye sahip olacaktır. Bu nedenle söz konusu platformun imhası düşman kuvvet için büyük bir kayıp olarak nitelendirilirken dost kuvvet için daha büyük kazanç oluşturacaktır. Bu nedenle 1'i tehdit denizaltı tespit olasılıklarına bölünerek ödül değeri ile çarpılmış ve daha küçük tespit olasılığına sahip bir denizaltının imhası ile daha yüksek ödül alınması sağlanmıştır. Bu kapsamda denizaltı tespit olasılıklarının belirlenmesinde literatürde yer alan ve aktif olarak kullanılan aşağıdaki formüllerden faydalanılmıştır [26].

$$p_i^j = 1 - e^{-\left[\frac{(w_i * l)}{a}\right]} \quad (3.1)$$

$$a = \pi * (v_j * t_j)^2 \quad (3.2)$$

$$l = v_i * t_j \quad (3.3)$$

p_i^j : i dost platformun j tehdit platformu tespit etme olasılığı.

w_i : i dost platformun sonar menzili.

v_i : i dost platformun sürati.

v_j : j tehdit platformun sürati.

t_j : j tehdit platformun sualtında kalma süresi.

Anılan formüller kullanılarak her bir dost kuvvet platformunun her bir tehdit denizaltıyı tespit etme olasılıkları hesaplanmıştır. Söz konusu hesaplama sonuçları Tablo 17’de sunulmuştur. Bununla birlikte dost kuvvet platformları denizaltı, suüstü ve hava unsurları olmak üzere karışık bir yapıdadır. Bu nedenle tehdit denizaltıların dost kuvvet unsurlarını tespit etme olasılıklarına ilişkin net bir hesaplama yapılamayacaktır. Bu nedenle her bir tehdit denizaltının her bir dost kuvvet platformunu tespit etmek olasılığı eşit olarak dikkate alınmıştır.

Tablo 17: Tehdit Denizaltı Tespit Olasılıkları.

Dost Platform \ Düşman Denizaltı	D/A Tip 1	D/A Tip 2	D/A Tip 3	D/A Tip 4	D/A Tip 5
Fırkateyn Tip 1	0,086	0,053	0,045	0,025	0,017
Fırkateyn Tip 2	0,111	0,069	0,058	0,033	0,022
Fırkateyn Tip 3	0,095	0,059	0,050	0,028	0,019
Korvet	0,082	0,051	0,043	0,024	0,016
Helikopter Tip1	0,379	0,251	0,216	0,129	0,087
Helikopter Tip2	0,368	0,243	0,209	0,124	0,084
Deniz Karakol Uçağı	0,537	0,373	0,325	0,200	0,138
Denizaltı Tip1	0,074	0,046	0,038	0,022	0,014
Denizaltı Tip2	0,735	0,045	0,038	0,021	0,014
Denizaltı Tip3	0,090	0,056	0,047	0,027	0,018
Denizaltı Tip4	0,066	0,041	0,034	0,019	0,013

BÖLÜM IV

EN İYİ HAREKAT TARZI VE PLATFORM/MÜHİMMAT SAYILARININ HESAPLANMASI

Çalışmanın buraya kadar olan bölümünde harekât tarzının belirlenmesi aşamasında girdi oluşturacak parametreler hesaplanmıştır. Bu bölümde ise önceki bölümlerde hesaplanan değerler kullanılarak optimal harekât tarzı belirlenecektir. Bu kapsamda pekiştirmeli öğrenme algoritmalarından faydalanılacaktır. Söz konusu algoritma ve hesaplamalara ilişkin detaylar müteakip maddelerde açıklanmıştır.

4.1 PEKİŞTİRMELİ ÖĞRENME ALGORİTMASININ İNCELENMESİ

Pekiştirmeli öğrenme, bir yapay zekâ alanı olan makine öğrenmesinin bir alt dalıdır. Bu öğrenme türü, bir sistem veya ajanın çevresiyle etkileşimde bulunarak deneyimlerinden öğrenmesini sağlar. Pekiştirmeli öğrenme, belirli bir görevde belirli bir performansı en üst düzeye çıkarmak için sistem tarafından alınan aksiyonların sonuçlarına dayanır. Öğrenme süreci, ajanın çevresiyle olan etkileşimlerini optimize etmeye çalıştığı bir döngüden oluşur [66, 67, 68, 69, 70].

Pekiştirmeli öğrenme süreci genellikle şu adımları içerir: durum gözlemleme, aksiyon seçme, çevreye etki etme ve elde edilen ödüllerin değerlendirilmesi. Ajan, çevresiyle etkileşimde bulunarak çeşitli durumları deneyimler ve bu deneyimlere dayanarak en uygun aksiyonları seçmeyi öğrenir. Ajanın amacı, belirli bir görevde en yüksek toplam ödülü elde etmektir [66, 67, 68, 69, 70].

Pekiştirmeli öğrenme algoritmaları genellikle Markov Karar Süreci (Markov Decision Process, MDP) çerçevesinde formüle edilir. MDP, bir dizi durum, aksiyon, ödül fonksiyonu ve geçiş olasılıklarını içeren bir matematiksel modeldir. Bu model, ajanın kararlarını ve etkileşimlerini tanımlamak için kullanılır [66, 67, 68, 69, 70].

Temel pekiştirmeli öğrenme algoritmalarından biri Q-öğrenme'dir. Q-öğrenme, ajanın her durum ve aksiyon çifti için bir değer fonksiyonu öğrenmesini sağlar. Bu fonksiyon, bir durumda belirli bir aksiyonun ne kadar "iyi" olduğunu temsil eder. Ajan, bu değer fonksiyonunu kullanarak en yüksek ödülü elde etmek için aksiyonları seçer [66, 67, 68, 69, 70].

Pekiştirmeli öğrenme algoritmalarının bir diğer önemli özelliği de keşfetme ve sömürme dengesini kurabilme yetenekleridir. Ajanın daha önce keşfetmediği ancak potansiyel olarak yüksek ödüller sunan aksiyonları keşfetme ihtiyacı vardır, ancak aynı zamanda daha önce öğrenilmiş ve iyi performans gösteren aksiyonları da sömürme eğilimindedir [66, 67, 68, 69, 70].

Özellikle, derin öğrenme tekniklerinin pekiştirmeli öğrenme ile birleştirilmesi, daha karmaşık ve büyük ölçekli problemleri çözmek için kullanılan derin pekiştirmeli öğrenme (deep reinforcement learning) alanını ortaya çıkarmıştır. Bu yaklaşım, genellikle büyük veri setleri ve karmaşık görevlerle başa çıkmak için güçlü öğrenme yeteneklerinden yararlanır [66, 67, 68, 69, 70].

Pekiştirmeli öğrenme algoritmaları geniş bir uygulama alanına sahiptir. Örneğin, oyun teorisi, robot kontrolü, finansal modelleme ve yapay zeka tabanlı karar verme gibi birçok alanda kullanılmaktadır. Bu algoritmalar, deneme-yanılma yöntemleriyle öğrenme yeteneği sayesinde karmaşık ve dinamik çevrelerde etkili bir şekilde kullanılabilir [66, 67, 68, 69, 70].

Bu nedenle DSH'nin modellenmesinde pekiştirmeli öğrenme (Q Learning) algoritmasının etkili çözümler geliştireceği değerlendirildiğinden bu çalışmada da Q Learning algoritmasından faydalanılmıştır. Çalışma kapsamında tasarlanan Q-Öğrenme algoritmasına ait sözde kod (pseudo-code) EK-2'de sunulmuştur.

4.2 ÖDÜL/CEZA DEĞERLERİNİN HESAPLANMASI

Yukarıda da açıklandığı üzere pekiştirmeli öğrenme algoritmaları bir ödül değeri hesaplamakta ve toplam ödül değerini optimize etmeye çalışmaktadır. Bu kapsamda bu çalışmada ödül değerleri platform ve silah maliyetleri göz önünde bulundurularak hesaplanmıştır. Söz konusu hesaplamalarda imha edilen platformun maliyeti ödül getirecekken angajmana girecek platformun maliyeti ve atılacak silah sayıları ceza olarak dikkate alınacaktır. Bu sayede bizim için daha yüksek maliyetli dolayısıyla daha değerli platformların öncelikli olarak angajmana girmesi engellenerek daha az değerli platformlarca angajman gerçekleştirilecek şekilde bir

planlamaya yapılabilmesi konusunda karar desteği sağlanacaktır. Ayrıca angajmanlar daha az maliyetli silahla gerçekleştirileceği için DSH planlamasının maliyet etkin bir şekilde yapılması sağlanabilecektir. Ayrıca denizaltılarının tespit olasılıkları da ödül/ceza hesaplamalarında kullanılmaktadır. Bu kapsamda 1'i düşman denizaltılarının tespit olasılıklarına bölerek daha az tespit olasılığına sahip bir denizaltının imhası ile daha fazla ödül değeri kazanılması sağlanmıştır. Bu doğrultuda ödül/ceza değerlerinin hesaplanmasında aşağıdaki formüllerden faydalanılmaktadır. Çalışma kapsamında olası imha edilecek dost kuvvet platformlarının belirlenebilmesi de amaçlandığından her iki tarafın hareket tarzı da belirlenmeye çalışılmaktadır. Ancak tehdit kuvvet için yapılan hesaplamalarda tespit olasılığı değeri dikkate alınmamaktadır.

$$r_k^d = \left[\left(c_k^{td} * \frac{1}{p_i^j} \right) - \left(c_k^d + (c_k^{dt} * a_k^{dt}) \right) \right] \quad (4.1)$$

$$r_k^t = \left[(c_k^d) - \left(c_k^{td} + (c_k^{tt} * a_k^{tt}) \right) \right] \quad (4.2)$$

r_k^d : k durumunda dost kuvvetin kazandığı ödül değeri.

c_k^{td} : k durumunda imha edilen düşman kuvvet D/A maliyeti.

c_k^d : k durumunda imha edilen dost kuvvet platform maliyeti.

c_k^{dt} : k durumunda imha edilecek D/A'nın imhası için dost kuvvet tarafından atılan silahın maliyeti.

a_k^{dt} : k durumunda imha edilecek D/A'nın imhası için dost kuvvet tarafından atılan silah sayısı.

r_k^t : k durumunda düşman kuvvetin kazandığı ödül değeri.

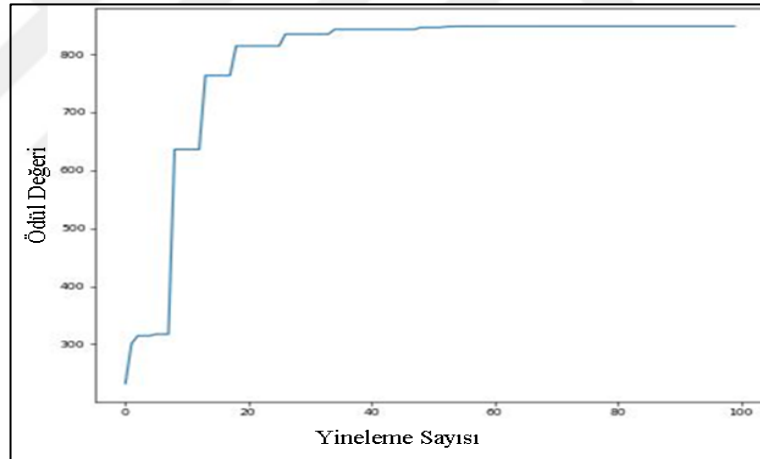
c_k^{tt} : k durumunda imha edilecek dost kuvvet platformunun imhası için düşman kuvvet tarafından atılan silahın maliyeti.

a_k^{tt} : k durumunda imha edilecek dost kuvvet platformunun imhası için düşman kuvvet tarafından atılan silah sayısı.

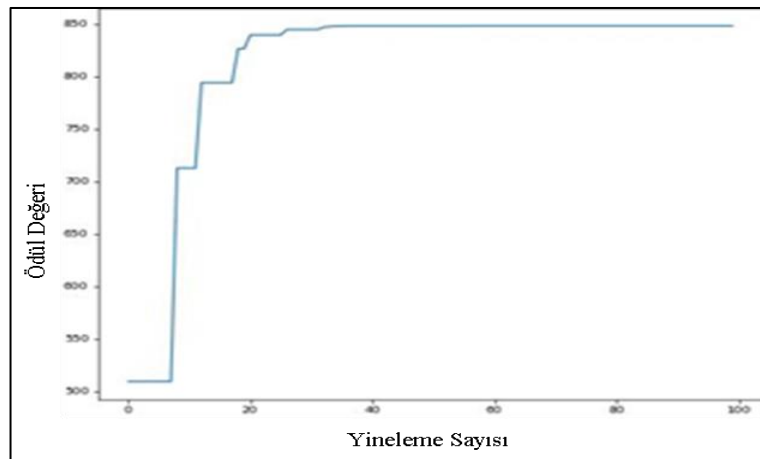
4.3 HİPERPARAMETRE OPTİMİZASYONU

Pekiştirmeli öğrenme algoritmalarının başarılı bir şekilde uygulanabilmesi için hiperparametre optimizasyonu önemli bir rol oynar. Hiperparametreler,

algoritmanın performansını etkileyen, ancak eğitim süreci içinde öğrenilmeyen parametrelerdir. Bu hiperparametrelerin doğru bir şekilde ayarlanması, algoritmanın istenilen görev üzerinde en iyi performansı göstermesini sağlar. Bir pekiştirmeli öğrenme algoritması için tipik hiperparametreler arasında yineleme sayısı, öğrenme oranı ve keşif oranı gibi değerler bulunur. Bu hiperparametreler, algoritmanın öğrenme hızını, görevi keşfetme yeteneğini ve ödülün zaman içindeki etkisini belirler. Hiperparametre optimizasyonu genellikle deneme-yanılma yöntemleri, rastgele arama, genetik algoritmalar, örnek tabanlı optimizasyon ve otomatik makine öğrenmesi araçları gibi tekniklerle gerçekleştirilir. Bu yöntemler, geniş bir hiperparametre uzayında gezinerek, algoritmanın en iyi performansı gösterdiği hiperparametre kombinasyonlarını belirlemeye çalışır [71]. Bu çalışmada deneme yanılma yöntemiyle bir optimizasyon gerçekleştirilmiştir. Yapılan incelemeler neticesinde yineleme sayısı-ödül değeri değişimini gösteren grafikler Şekil (52 ve 53)'te sunulmuştur.

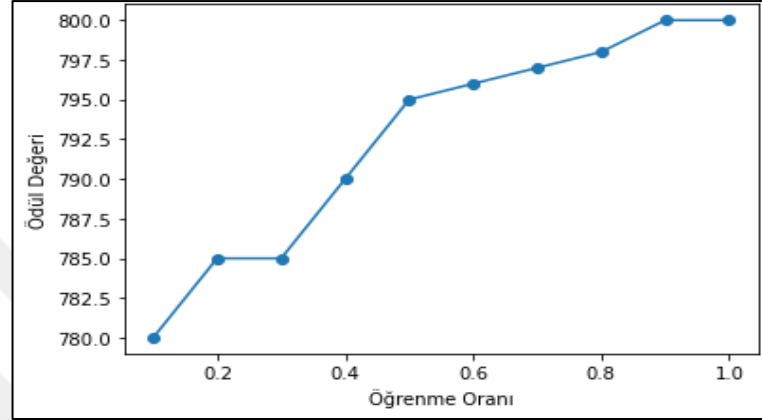


Şekil 52: Dost Kuvvet Yineleme Sayısı-Ödül Değeri Grafiği.

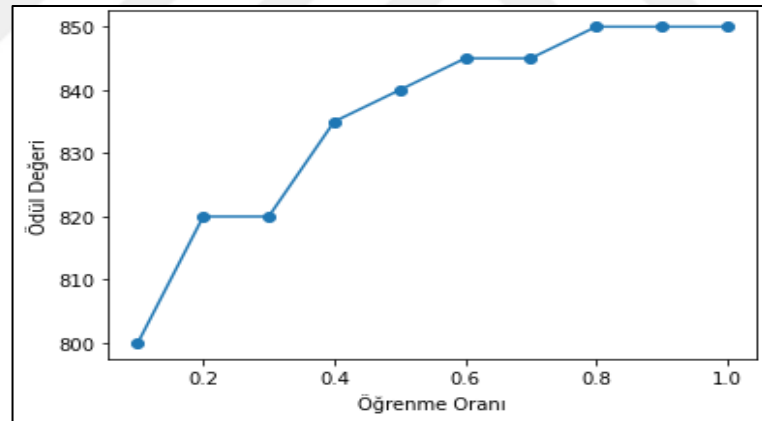


Şekil 53: Düşman Kuvvet Yineleme Sayısı-Ödül Değeri Grafiği.

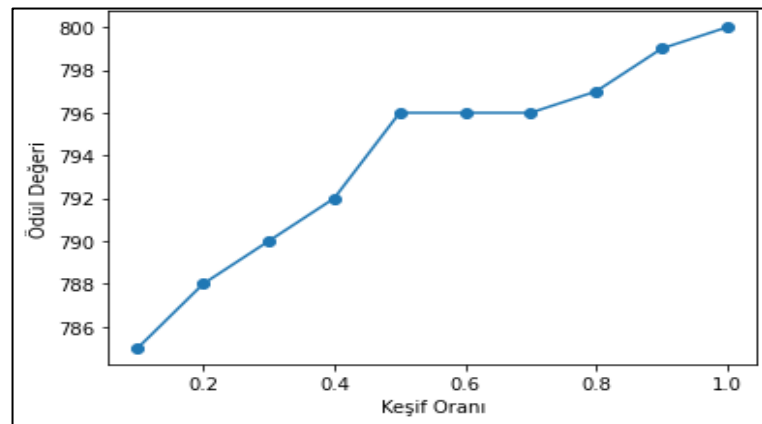
Şekil (52 ve 53)'te de görüldüğü üzere yineleme sayısı 35 üzerine çıktıktan sonra ödül değerinde çok büyük değişimler yaşanmamaktadır. Bu nedenle çalışma kapsamında yineleme sayısı 35 olarak hesaplamaya dahil edilmiştir. Bununla birlikte öğrenme oranı ve keşif oranı içinde benzer hesaplamalar yapılmıştır. Bahse konu hesaplama sonuçları Şekil (54-57)'de sunulmuştur.



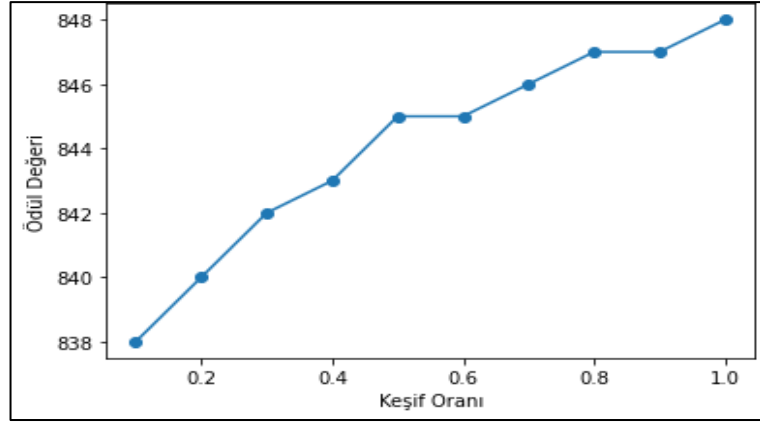
Şekil 54: Dost Kuvvet Öğrenme Oranı-Ödül Değeri Grafiği.



Şekil 55: Düşman Kuvvet Öğrenme Oranı-Ödül Değeri Grafiği.



Şekil 56: Dost Kuvvet Keşif Oranı-Ödül Değeri Grafiği.



Şekil 57: Düşman Kuvvet Keşif Oranı-Ödül Değeri Grafiği.

Şekil (54-57)'de yer alan veriler incelendiğinde öğrenme oranı ve keşif oranındaki değişimin ödül değerinde anlamlı bir fark oluşturmadağı görülmektedir. Bu nedenle çalışmada her iki değerin 0,5 olarak hesaplamaaya dahil edilmesinin uygun olacağı değerlendirilmiştir.

Ayrıca hesaplamalarda bazı kabuller de yapılmıştır. Bu kapsamda;

- Tüm angajmanların T0 anında gerçekleşeceği kabul edilmiştir.
- Tüm platformların angajman menzili içerisinde olduğu kabul edilmiştir.
- Bir hedefe herhangi bir platform/denizaltı angajman gerçekleştirmiş ise başka bir platform/denizaltının angajman gerçekleştirmeyeceği kabul edilmiştir.

4.4 OPTİMAL HAREKÂT TARZININ BELİRLENMESİ

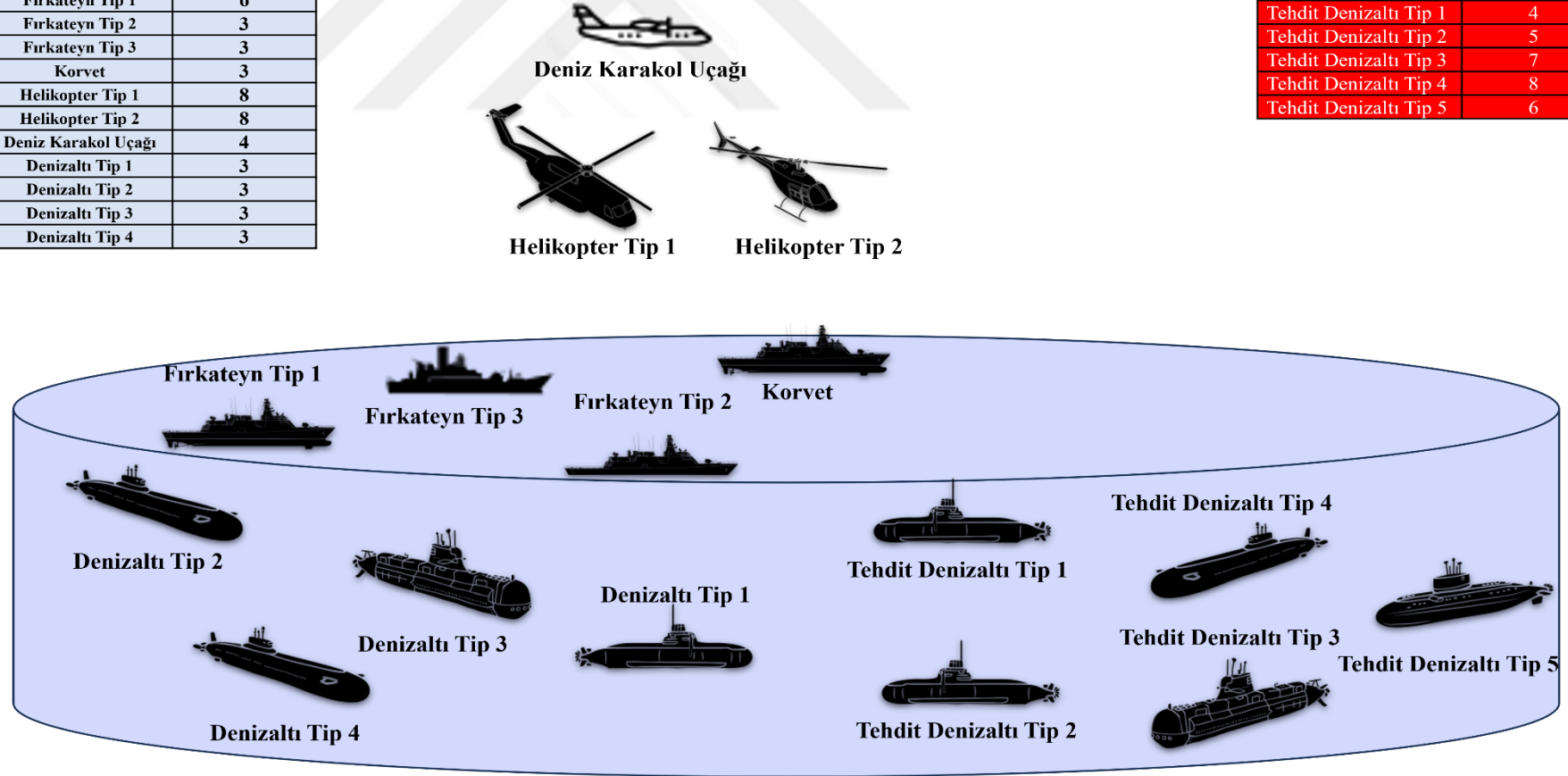
Önceki maddeler açıklanan/hesaplanan veriler doğrultusunda Tablo 18 ve Şekil 54'te sunulan bir harekât ortamı için optimal harekât tarzı belirlenmiştir. Söz konusu hesaplama sonuçları Tablo (19 ve 20)'de sunulmuştur.

Tablo 18: Hesaplama Kullanan Platform Bilgileri.

Dost Kuvvet		Düşman Kuvvet	
Platform	Miktar	Platform	Miktar
Fırkateyn Tip 1	6	Tehdit Denizaltı Tip 1	4
Fırkateyn Tip 2	3	Tehdit Denizaltı Tip 2	5
Fırkateyn Tip 3	3	Tehdit Denizaltı Tip 3	7
Korvet	3	Tehdit Denizaltı Tip 4	8
Helikopter Tip 1	8	Tehdit Denizaltı Tip 5	6
Helikopter Tip 2	8		
Deniz Karakol Uçağı	4		
Denizaltı Tip 1	3		
Denizaltı Tip 2	3		
Denizaltı Tip 3	3		
Denizaltı Tip 4	3		

Dost Kuvvet	
Platform Adı	Miktar
Fırkateyn Tip 1	6
Fırkateyn Tip 2	3
Fırkateyn Tip 3	3
Korvet	3
Helikopter Tip 1	8
Helikopter Tip 2	8
Deniz Karakol Uçağı	4
Denizaltı Tip 1	3
Denizaltı Tip 2	3
Denizaltı Tip 3	3
Denizaltı Tip 4	3

Düşman Kuvvet	
Platform Adı	Miktar
Tehdit Denizaltı Tip 1	4
Tehdit Denizaltı Tip 2	5
Tehdit Denizaltı Tip 3	7
Tehdit Denizaltı Tip 4	8
Tehdit Denizaltı Tip 5	6



Şekil 58: Hesaplama Dikkate Alınan Harekât Ortamının Görselleştirilmesi.

Tablo 19: Dost Kuvvet Optimal Harekât Tarzı.

Atıcı Platform	Hedef Platform
Helikopter Tip 1-6	D/A Tip1-1
Fırkateyn Tip3-1	D/A Tip1-2
Fırkateyn Tip2-3	D/A Tip1-3
Denizaltı Tip3-1	D/A Tip1-4
Fırkateyn Tip1-4	D/A Tip2-1
Fırkateyn Tip2-2	D/A Tip2-2
Helikopter Tip1-7	D/A Tip2-3
Helikopter Tip2-7	D/A Tip2-4
Helikopter Tip2-1	D/A Tip2-5
Helikopter Tip2-4	D/A Tip3-1
Denizaltı Tip3-3	D/A Tip3-2
Helikopter Tip1-4	D/A Tip3-3
Helikopter Tip2-3	D/A Tip3-4
Helikopter Tip1-8	D/A Tip3-5
Helikopter Tip1-7	D/A Tip3-6
Korvet-1	D/A Tip3-7
Korvet-2	D/A Tip4-1
Deniz Karakol Uçağı-4	D/A Tip4-2
Fırkateyn Tip1-6	D/A Tip4-3
Helikopter Tip2-8	D/A Tip4-4
Fırkateyn Tip3-2	D/A Tip4-5
Deniz Karakol Uçağı-2	D/A Tip4-6
Helikopter Tip2-4	D/A Tip4-7
Deniz Karakol Uçağı-1	D/A Tip4-8
Korvet-1	D/A Tip5-1
Denizaltı Tip2-1	D/A Tip5-2
Deniz Karakol Uçağı-3	D/A Tip5-3
Denizaltı Tip2-2	D/A Tip5-4
Fırkateyn Tip1-1	D/A Tip5-5
Korvet-2	D/A Tip5-6

Tablo 20: Düşman Kuvvet Optimal Harekât Tarzı.

Atıcı Platform	Hedef Platform
D/A Tip4-5	Fırkateyn Tip1-2
D/A Tip3-6	Fırkateyn Tip1-4
D/A Tip2-3	Fırkateyn Tip1-5
D/A Tip3-5	Fırkateyn Tip1-1
D/A Tip4-6	Fırkateyn Tip1-6
D/A Tip5-2	Fırkateyn Tip2-1
D/A Tip3-2	Fırkateyn Tip3-2
D/A Tip4-8	Fırkateyn Tip3-3
D/A Tip3-4	Korvet-1
D/A Tip5-6	Korvet-2
D/A Tip3-7	Korvet-4
D/A Tip5-5	Helikopter Tip1-8
D/A Tip4-3	Helikopter Tip2-5
D/A Tip2-1	Helikopter Tip2-7
D/A Tip4-1	Deniz Karakol Uçağı-1
D/A Tip5-3	Deniz Karakol Uçağı-3
D/A Tip2-5	Denizaltı Tip1-1

Tablo 20'nin Devamı.

Atıcı Platform	Hedef Platform
D/A Tip4-4	Denizaltı Tip1-2
D/A Tip5-1	Denizaltı Tip2-1
D/A Tip1-2	Denizaltı Tip3-3
D/A Tip4-7	Denizaltı Tip3-2
D/A Tip5-3	Denizaltı Tip3-1
D/A Tip3-1	Denizaltı Tip4-1
D/A Tip4-2	Denizaltı Tip4-2

Tablo (19 ve 20)'de yer alan sonuçlar incelendiğinde modelin istenilen şekilde daha az değerli platformlar ile angajman gerçekleştirecek, daha değerli platformların ise neredeyse hiç angajmana girmeyecek şekilde bir planlama yaptığını görebiliriz. Düşman kuvvet için hesaplanan angajman planı incelendiğinde de daha çok yine dost kuvvetin değerli platformlarına yönelik bir angajman planlaması yapıldığı görülmektedir. Bu bilgiler doğrultusunda değerli platformların hareket sahasında bulundurulmaması hem değerli platformların korunmasını hem de DSH'ın dost kuvvetin lehine sonuçlanmasını sağlayacaktır.

4.5 TOPLAM MÜHİMMAT SAYILARININ HESAPLANMASI

Önceki bölümde optimal angajman planı belirlenmiştir. Bu bölümde ise söz konusu angajman planına istinaden ihtiyaç duyulacak toplam mühimmat sayıları belirlenmiştir. Bu dost kuvvetin için Tablo 19'daki hareket tarzı için ihtiyaç duyacağı toplam mühimmat sayıları Tablo 21'de sunulmuştur.

Tablo 21: Toplam Mühimmat Stok Miktarları.

Mühimmat Adı	Toplam İhtiyaç Miktarı
T1	38
T2	51
T3	17

BÖLÜM V

SONUÇ VE DEĞERLENDİRME

5.1 MEVCUT YAKLAŞIMLARIN İNCELENMESİ

Halihazırda mühimmat stok planlamalarına yönelik olarak tehdiye yönelik olasılıksal yaklaşım ile hesaplamalar yapılmaktadır. Bahse konu yöntem literatürde yer alan hesaplama yöntemleri dikkate alınarak geliştirilmiştir. Ancak hesaplama yönteminin detayları gizlilik dereceleri bilgiler içerdiğinden bu çalışmada paylaşılamamaktadır. Bununla birlikte temelde belirli formüller kullanılarak atılması gereken mühimmat sayıları hesaplanmakta, ardından uzman görüşü, personel tecrübesi vb. subjektif kabullere istinaden nihai stok miktarları hesaplanmaktadır. Bu kapsamda bu çalışmada hesaplanan atılması gereken mühimmat sayıları dikkate alınarak tehdiye yönelik olasılıksal yaklaşım ile toplam stok miktarları hesaplanmıştır. Elde edilen hesaplama sonuçlarına ilişkin detaylar Tablo 22’de sunulmuştur.

Tablo 22: Tehdiye Yönelik Olasılıksal Yaklaşım ile Hesaplanan Toplam Mühimmat Stok Miktarları.

Mühimmat Adı	Toplam İhtiyaç Miktarı
T1	41
T2	74
T3	37

5.2 BU ÇALIŞMA İLE ELDE EDİLEN KAZANIMLAR

Önceki maddede de görüldüğü üzere tehdiye yönelik olasılıksal yaklaşım ile yapılan hesaplamalarda bu çalışma hesaplanan stok miktarlarına göre daha fazla mühimmat stok miktarı hesaplanmaktadır. Bu durum ilave maliyete katlanılmasına neden olmaktadır. Bu hesaplamada dikkate alınan birim maliyetlere göre tehdiye yönelik olasılıksal yöntem ile ortaya çıkan toplam maliyet 299,5 milyon dolar olurken, bu çalışma hesaplanan stok miktarlarına göre toplam maliyet 206,1 milyon

dolar olmaktadır. Bu da yaklaşık olarak 93,4 milyon dolarlık bir tasarrufa imkân sağlamaktadır.

Ayrıca tehdide yönelik olasılıksal yaklaşım ile sadece mühimmat stok miktarları hesaplanabilmektedir. Bu çalışma ile mühimmat stok miktarlarının yanında aynı zamanda DSH harekâtı kapsamında kullanılacak platform miktarları ve DSH harekâtına yönelik bir hareket planlaması konusunda da karar desteği sağlanabilmektedir. Bu sayede angajmana girmeyecek platformlar farklı harp nevelerinde kullanılmak üzere planlanabilmektedir. Ayrıca model temelde değerli platformların angajmana girmesinin önüne geçtiğinden yüksek değerlikli platformların korunmasını da sağlayacaktır.

Tehdide yönelik olasılıksal yaklaşımda hesaplamalar uzman görüşü, personel tecrübesi vb. subjektif kabullere göre yapılmaktadır. Bu durum personel değişikliği ile değişiklik göstermektedir. Buda farklı dönemlerde yapılan analizlerde farklılıklar yaşanmasına sebep olabilmektedir. Bu çalışmada ise tamamen veriye dayalı bir yaklaşım söz konusudur. Bu nedenle hesaplama sonuçları subjektif değerlerden bağımsız ve tutarlı bir şekilde yapılabilmektedir.

5.3 MÜTEAKİP DÖNEMDE YAPILABİLECEK ÇALIŞMALAR

Bu çalışma ile Deniz Kuvvetleri Komutanlığı envanterindeki DSH kapsamında kullanılacak platformların ve platformlar üzerindeki silahların optimal seviyede planlanması için hesaplamalar yapılmıştır. Ancak herhangi bir deniz savaşında sadece DSH için planlama yapmak yeterli olmayacaktır. Suüstü Harbi (SUH) ve Elektronik Harp gibi harp neveleri için de planlamalar yapılması gerekecektir. Bu nedenle bu çalışmanın tüm harp nevelerini içerecek şekilde genişletilmesi ile daha etkin bir planlama aracına sahip olunabilecektir. Ayrıca söz konusu çalışmaların simülasyon yetenekleri ile birleştirilmesi ile daha gerçekçi sonuçlar elde edilebilecektir. Bununla birlikte bu çalışmada otonom araçlar dikkate alınmamıştır. Çalışmanın otonom araçları da içerecek şekilde güncellenmesi ile daha doğru bir planlama yapılacaktır. Bu kapsamda yukarıda açıklanan hususları içerecek şekilde çalışma kapsamının genişletilebileceği değerlendirilmektedir.

KAYNAKÇA

- [1] SIĞIRCI Mehmet (2020), *Denizaltı: Kim, Ne Zaman İcat Etti?*, <https://bilimgenc.tubitak.gov.tr/makale/denizaltinin-icadi>, ET. 20.11.2023.
- [2] KARACA Erkan (2016), *Cornelius Van Drebbel Kimdir? Hayatı ve Eserleri*, <https://bilgihanem.com/cornelius-van-drebbel-kimdir/>, ET.20.11.2023.
- [3] DEMİRTAŞ Ali Osman (2020), *Hem Suyun Üstünde Hem Altında Yüzen İcat: Denizaltı*, <https://muhendisgelisim.com/hem-suyun-ustunde-hem-altinda-yuzen-icat-denizalti/>, ET.20.11.2023.
- [4] KIRIL Ryabov (2020), *Denizaltı HL Hunley. Suudi Arabistan'ın trajik deneyimi*, <https://tr.topwar.ru/172977-podlodka-hl-hunley-tragicheskij-opyt-ksha.html>, ET. 20.11.2023.
- [5] Su Üstünde (2023), *USS Monitor ve H.L. Hunley*, <https://www.suustunde.com/tr/content-details/uss-monitor-ve-hl-hunley-.html?ContentID=617>, ET.20.11.2023.
- [6] VİKİPEDİ (2023), *U-Bot*, <https://tr.wikipedia.org/wiki/U-bot>, ET. 20.11.2023.
- [7] VİKİPEDİ (2023), *U-bot Filotillası*, https://tr.wikipedia.org/wiki/30._U-bot_Filotillas%C4%B1, ET. 20.11.2023.
- [8] MİSTİK ALEM (2022), *Nazilerin Hayalet Denizaltıları: U-botlar*, <https://www.mistikalem.com/xfiles/nazilerin-hayalet-denizaltıları-u-botlar-haberi-19037>, ET. 23.11.2023.
- [9] VİKİPEDİ (2023), *German submarine U-9*, [https://en.wikipedia.org/wiki/German_submarine_U-9_\(1935\)](https://en.wikipedia.org/wiki/German_submarine_U-9_(1935)), ET. 20.11.2023.
- [10] AYTAN Deniz (2007), *İkinci Dünya Savaşı Atlantik Deniz Muharabelerinde Denizaltının Rolü*, https://acikbilim.yok.gov.tr/bitstream/handle/20.500.12812/339542/yokAcikBilim_301148.pdf?sequence=-1&isAllowed=y, ET. 24.10.2023.

- [11] SCHERER Scott D. (2009), *Game-Theoretic Anti-Submarine Warfare Mission Planner (Heuristic-Based, Fully Excel Capable)* (Yüksek Lisans Tezi), Naval Postgraduate School, Monterey, California.
- [12] KIM Roger G. (2017), *Operational Planning For Theater Anti-Submarine Warfare* (Yüksek Lisans Tezi), Naval Postgraduate School, Monterey, California.
- [13] TIMMERMAN Michael Jay (1980), *A Genetic Algorithm Based Anti-Submarine Warfare Simulator* (Yüksek Lisans Tezi), Naval Postgraduate School, Monterey, California.
- [14] YOASH Roey Ben (2016), *Anti-Submarine Warfare Search Models* (Yüksek Lisans Tezi), Naval Postgraduate School, Monterey, California.
- [15] LUO Xinwei, CHEN Lu, ZHOU Hanlu ve CAO Hongli (2023), "A Survey of Underwater Acoustic Target Recognition Methods Based on Machine Learning", *Journal of Marine Science and Engineering*, Cilt 11, No 2, ss. 384-400.
- [16] I. B. SAKSVIK, A. ALCOCER ve V. HASSANI (2021), *A Deep Learning Approach To Dead-Reckoning Navigation For Autonomous Underwater Vehicles With Limited Sensor Payloads*, <https://www.semanticscholar.org/reader/f4a6f8b309208bba643004c318f2753eb77fcfa6>, ET. 29.11.2023.
- [17] JIN Anqi ve ZENG Xiangyang (2023), "A Novel Deep Learning Method for Underwater Target Recognition Based on Res-Dense Convolutional Neural Network with Attention Mechanism", *Journal of Marine Science and Engineering*, Cilt 11, No 1, ss. 69-88.
- [18] HSU Yachu, WU Hui, YOU Keyou, SONG Shijia (2019), "Selected Review On Reinforcement Learning Based Control For Autonomous Underwater Vehicles", *Scientia Sinica Informations*, Cilt 50, No 22, ss. 1798.
- [19] HOZYN Stanisław (2021), "A Review of Underwater Mine Detection and Classification in Sonar Imagery", *Electronics* Cilt 10, No 23, Makale Numarası 2943, DOI: 10.3390/electronics10232943.

- [20] YANG Haesang, LEE Keunhwa, CHOO Youngmin ve KİM Kookhyun (2020), "Underwater Acoustic Research Trends with Machine Learning: General Background", *Journal of Ocean Engineering and Technology*, Cilt 34, No 2, ss. 147-154.
- [21] JACINTO Edwar (2023), "Navigation of Autonomous Vehicles using Reinforcement Learning", *International Journal of Advanced Computer Science and Applications*, Cilt 14, No 1, ss. 954-959.
- [22] AKBORİ Fahrettin (2004), *Autonomous-Agent Based Simulation of Anti-Submarine Warfare Operations with the Goal of Protecting a High Value Unit* (Yüksek Lisans Tezi), Naval Postgraduate School, Monterey, California.
- [23] ÜNLÜ Salim (2015), *Effectiveness of Unmanned Surface Vehicles in Anti-Submarine Warfare with the Goal of Protecting a High Value Unit* (Yüksek Lisans Tezi), Naval Postgraduate School, Monterey, California.
- [24] STEPHEN JR F. Fahey (2016), *Software Architecture for Anti-Submarine Warfare Unmanned Surface Vehicles* (Yüksek Lisans Tezi), Naval Postgraduate School, Monterey, California.
- [25] Operations Analysis Study Group (1989), *Naval Operations Analysis*, Naval Institute Press, Annapolis, Maryland, ss. 127.
- [26] VİKİPEDİ (2023), 214 Sınıfı Denizaltı, https://tr.wikipedia.org/wiki/214_s%C4%B1n%C4%B1f%C4%B1_denizalt%C4%B1, ET. 24.10.2023.
- [27] VİKİPEDİ (2023), ADA Sınıfı Korvet, https://tr.wikipedia.org/wiki/Ada_s%C4%B1n%C4%B1f%C4%B1_korvet, ET. 24.11.2023.
- [28] VİKİPEDİ (2023), Sikorsky UH-60 Black Hawk, https://tr.wikipedia.org/wiki/Sikorsky_UH-60_Black_Hawk, ET. 24.10.2023.
- [29] VİKİPEDİ (2023), Lockheed P-3 Orion, https://tr.wikipedia.org/wiki/Lockheed_P-3_Orion, ET. 24.10.2023.
- [30] VİKİPEDİ (2023), FDI Belharra Sınıfı Firkateyn, https://tr.wikipedia.org/wiki/FDI_Belharra_s%C4%B1n%C4%B1f%C4%B1_f%C4%B1rkateyn, ET. 24.10.2023.

- [31] SÜNNETÇİ İbrahim (2017), *AKYA Milli Ağır Torpidosu ile Hedefe İlk Atış Testi Başarıyla İcra Edildi*, <https://www.defenceturkey.com/tr/icerik/akya-milli-agir-torpidosu-ile-hedefe-ilk-gudumlu-atis-testi-basariyla-icra-edildi-4377>, ET. 24.10.2023.
- [32] E. KALIPÇI, V. BAŞER, M. TÜRKMEN, N. GENÇ ve H. CÜCE (2021), "Türkiye Kıyılarında Deniz Suyu Sıcaklık Değişiminin CBS ile Analizi ve Ekolojik Etkilerinin Değerlendirilmesi", *Doğal Afet ve Çevre Dergisi*, Cilt 7, No. 2, ss. 278-288.
- [33] KASKATI Mustafa Tuna (2010), *Kuzey Ege'nin Oşinografik Parametrelerinin Mevsimsel ve Yıllar Bazında İncelenmesi* (Yüksek Lisans Tezi), İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul.
- [34] ORAN Mustafa (1994), *Kuzey Ege Denizi Su Kütleleri ve Dağılım Özellikleri* (Doktora Tezi), İstanbul Üniversitesi Deniz Bilimleri ve İşletmeciliği Enstitüsü, İstanbul.
- [35] YALILI KILIÇ Melike, ADALI Sümeyye (2020), "Deniz Ulaşımından Kaynaklanan Gürültü Kirliliğinin Belirlenmesi: Bursa Güzelyalı Örneği", *Uludağ Üniversitesi Mühendislik Fakültesi Dergisi*, cilt 25, ss. 1015-1024.
- [36] AKSOY Serkan (2022), *Su Altı Akustik Dalga Yayılımı & Sonar Sistemleri* (Yüksek Lisans Tezi), Gebze Teknik Üniversitesi, Kocaeli.
- [37] ATA Özer (2015), *Gemi Pervanelerinde Kavite Kaynaklı Gürültünün İncelenmesi* (Yüksek Lisans Tezi), Yıldız Teknik Üniversitesi Gemi İnşaatı ve Gemi Makineleri Mühendisliği Ana Bilim Dalı, İstanbul.
- [38] MURATLAR Emre Rıdvan (2021), *Dengesiz Veri Setlerinde Modelleme*, <https://www.veribilimiokulu.com/dengesiz-veri-setlerinde-modelleme/>, ET. 24.10.2023.
- [39] GOSWAMI Saptarsi (2020), *Class Imbalance, SMOTE, Borderline SMOTE, ADASYN*, <https://towardsdatascience.com/class-imbalance-smote-borderline-smote-adasync-6e36c78d804>, ET. 24.10.2023.
- [40] HE Haibo, BAI Yang, GARCÍA Eduardo A., LI Shutao (2010), "ADASYN: Adaptive Synthetic Sampling Approach for Imbalanced Learning", *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, ss. 1322-1328, Hong Kong.

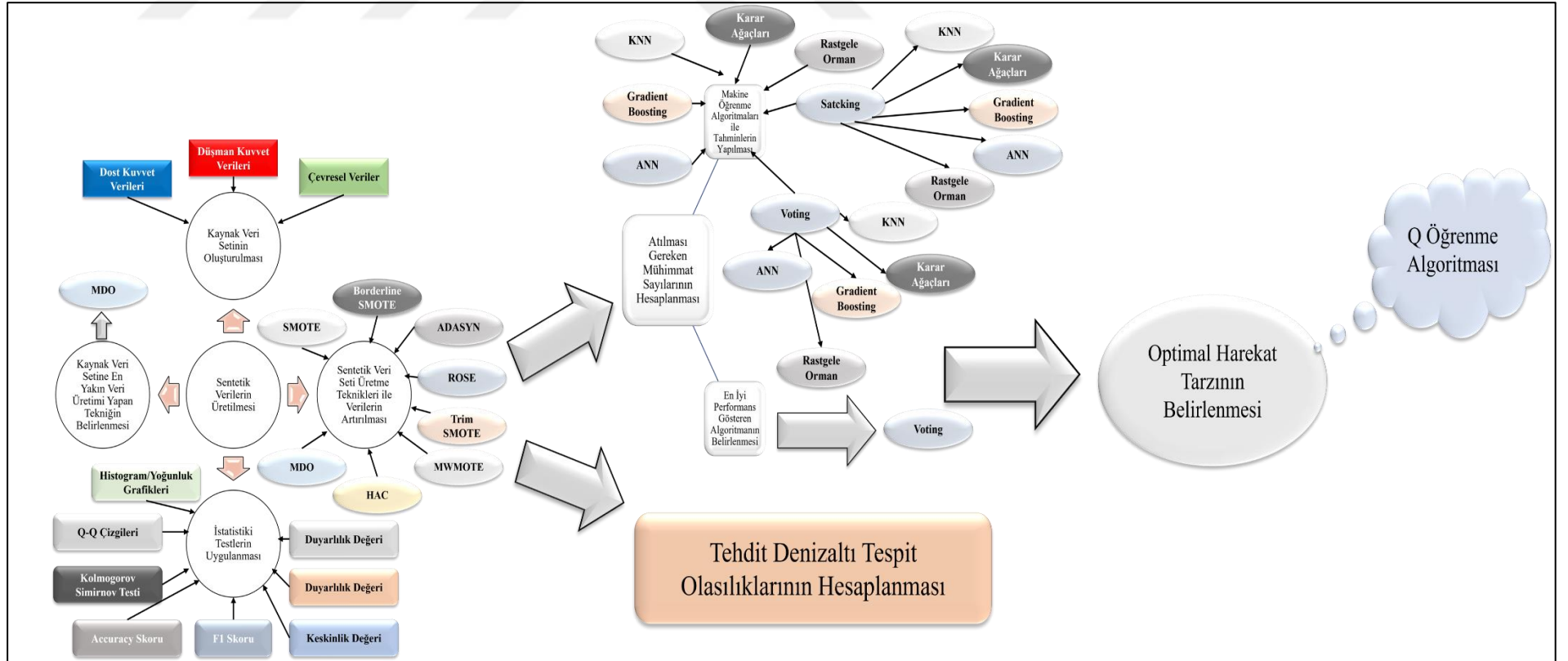
- [41] WISAM Essam (2023), *Class Imbalance: ROSE and Random Walk Oversampling (RWO)*, <https://towardsdatascience.com/class-imbalance-from-random-oversampling-to-rose-517e06d7a9b>, ET. 24.10.2023.
- [42] KOVACS Gyorgy (2019), "An Empirical Comparison And Evaluation Of Minority Oversampling Techniques On A Large Number Of Imbalanced Datasets", *Applied Soft Computing Journal*, cilt 83, ss. 1-13,
- [43] VERİ BÜLTENİ (2019), *Haftanın Grafiği 14 — Yoğunluk Grafiği*, <https://medium.com/veri-b%C3%BClteni/haftan%C4%B1n-grafi%C4%9Fi-14-yo%C4%9Funluk-grafi%C4%9Fi-7e296cfd8bfb>, ET. 23.11.2023.
- [44] IBM (2023), *Histogram grafikleri*, <https://www.ibm.com/docs/tr/watsonx-as-a-service?topic=types-histogram-charts>, ET. 23.11.2023.
- [45] VİKİPEDİ (2023), *Q-Q Plot*, https://en.wikipedia.org/wiki/Q%E2%80%93Q_plot, ET. 23.11.2023.
- [46] FORD Clay (2015), *Understanding QQ Plots*, <https://library.virginia.edu/data/articles/understanding-q-q-plots>, ET. 24.10.2023.
- [47] VİKİPEDİ (2022), *Kolmogorov-Smirnov Sinaması*, https://tr.wikipedia.org/wiki/Kolmogorov-Smirnov_s%C4%B1namas%C4%B1, ET. 24.10.2023.
- [48] ÖĞÜNDÜR Gülcan (2019), *Doğruluk (Accuracy) , Kesinlik(Precision) , Duyarlılık(Recall) ya da F1 Score?*, <https://medium.com/@gulcanogundur/do%C4%9Fruluk-accuracy-kesinlik-precision-duyarl%C4%B1l%C4%B1k-recall-ya-da-f1-score-300c925feb38>, ET. 24.10.2023.
- [49] DEVREYAKAN (2021), *Performans Metrikleri*, <https://devreyakan.com/performans-metrikleri/>, ET. 30.10.2023.
- [50] ÖĞÜNDÜR Gülcan (2020), *ROC ve AUC*, <https://medium.com/@gulcanogundur/roc-ve-auc-1fefcfc71a14>, ET. 24.10.2023.
- [51] MURPHY Kevin P. (2012), *Machine Learning: A Probabilistic Perspective*, The MIT Press Cambridge/Massachusetts, London, ss.1-24.

- [52] BISHOP Christopher M. (2006), *Pattern Recognition and Machine Learning*, Information Science and Statics, New York, ss. 4-39.
- [53] ATCILI Abdullah (2022), *KNN (K-En Yakın Komsu)*, <https://medium.com/machine-learning-t%C3%BCrkiye/knn-k-en-yak%C4%B1n-kom%C5%9Fu-7a037f056116>, ET. 23.11.2023.
- [54] TAFRALI Serdar (2022), *K-en yakın komşu algoritması nasıl çalışır?*, <https://miuul.com/not-defteri/k-en-yakin-komsu-algoritmasi-nasil-calisir>, ET. 23.11.2023.
- [55] BULUTİSTAN (2023), *Makine Öğrenmesi Karar Ağacı (Decision Tree) Nedir?*, <https://bulutistan.com/blog/makine-ogrenmesi-karar-agaci-decision-tree-nedir/>, ET. 23.11.2023.
- [56] AKÇA Mehmet Fatih (2020), *Karar Ağaçları (Makine Öğrenmesi Serisi-3)*, <https://medium.com/deep-learning-turkiye/karar-a%C4%9Fa%C3%A7lar%C4%B1-makine-%C3%B6%C4%9Frenmesi-serisi-3-a03f3ff00ba5>, ET. 23.11.2023.
- [57] DEV HUNTER (2018), *Rastgele Orman(Random Forest) Algoritması*, <https://devhunteryz.wordpress.com/2018/09/20/rastgele-ormanrandom-forest-algoritmasi/>, ET. 23.11.2023.
- [58] ÇEBİ Cem Berke (2020), *Rastgele Orman Algoritması*, <https://medium.com/@cemthecebi/rastgele-orman-algoritmas%C4%B1-1600ca4f4784>, ET. 23.11.2023.
- [59] DEV HUNTER (2018), *Gradyan Artırma (Gradient Boostin)*, <https://devhunteryz.wordpress.com/2018/07/11/gradyan-arttirmagradient-boosting/>, ET. 23.11.2023.
- [60] YILDIRIM Ecehan (2021), *Temel Topluluk Öğrenimi ve Random Forest, Gradient Boosting Algoritmaları*, <https://ekehanyildirim.medium.com/temel-topluluk-%C3%B6%C4%9Frenimi-ve-random-forest-gradient-boosting-algoritmalar%C4%B1-e105f93b33f0>, ET. 23.11.2023.
- [61] HOŞ Sibel (2023), *Neural Networks Nedir? Nasıl Çalışır?*, <https://www.hosting.com.tr/blog/neural-networks/>, ET. 23.11.2023.

- [62] AKDAĞLI Ece (2021), *Yapay Sinir Ağları (Artificial Neural Network (ANN))*, <https://ece-akdagli.medium.com/yapay-sinir-a%C4%9Flar%C4%B1-artificial-neural-network-ann-d7846dfcb022>, ET. 23.11.2023.
- [63] YILDIRIM Elif (2020), *Yapay Sinir Ağı(Artificial Neural Network) Nedir?*, <https://www.veribilimiokulu.com/yapay-sinir-agiartificial-neural-network-nedir>, ET. 23.11.2023.
- [64] GERON Aurelien (2019), *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow (Concepts, Tools and Techniques to Build Intelligent Systems)*, İkinci Basım, O Reilly USA, ss. 210-214.
- [65] KÖSEOĞLU Buse (2021), *Model Performansını Değerlendirmek: Regresyon*, <https://medium.com/yaz%C4%B1l%C4%B1m-ve-bili%C5%9Fim-kul%C3%BCb%C3%BC/model-performans%C4%B1n%C4%B1-de%C4%9Ferlendirmek-regresyon-48b4afec8664>, ET. 24.10.2023.
- [66] Pekiştirmeli Öğrenme Takımı (2019), *Pekiştirmeli Öğrenme - Bölüm 1: Giriş*, <https://yz-ai.github.io/blog/pekistirmeli-ogrenme/pekistirmeli-ogrenme-bolum-1>, ET. 24.11.2023.
- [67] SUTTON Richard S. ve BARTO Andrew G. (2014), *Reinforcement Learning: An Introduction*, İkinci Basım, The MIT Press, Cambridge London, ss.1-13.
- [68] RAO Ashwin ve JELVIS Tikhon (2022), *Foundations of Reinforcement Learning with Applications in Finance*, Birinci Basım, Chapman and Hall /CRC, ss. 17-33.
- [69] DOĞAN Özkan (2019), *Pekiştirmeli Öğrenmeye Giriş*, <https://medium.com/deep-learning-turkiye/peki%C5%9Ftirmeli-%C3%B6%C4%9Frenmeye-giri%C5%9F-c7c2a8cce50b>, ET. 24.11.2023.
- [70] AKDAĞLI Ece (2021), *Pekiştirmeli Öğrenme (Reinforcement Learning) Nedir*, <https://www.muendisbeyinler.net/pekistirmeli-ogrenme-reinforcement-learning-nedir/>, ET. 24.11.2023.
- [71] ÇARKACI Necmettin (2018), *Derin Öğrenme Uygulamalarında Hiper Parametre Seçim Yöntemleri*, <https://medium.com/deep-learning-turkiye/derin-ogrenme-uygulamalarinda-model-dogrulama-ve-hiper-parametre-secim-yontemleri-823812d95f3>, ET. 24.11.2023.

EKLER

EK-1 HESAPLAMA METODOLOJİSİNİN GÖSTERİMİ



EK-2 Q-ÖĞRENME ALGORİTMASINA AİT SÖZDE KOD (PSEUDO-CODE)

Önerilen Q-Öğrenme Algoritması

1. $R_{\text{önceki}}=0$
2. S ortam modeli=0
3. For $\rightarrow \delta$
4. If $\rightarrow \varepsilon > (0-1 \text{ arasında rasgele bir gerçel sayı})$
5. S matrisine 0 veya 1 olarak $\begin{pmatrix} 0 & \dots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \dots & 0 \end{pmatrix}$ rasgele değerler ata.
6. For $\rightarrow$ Her bir düşman platform için;
7. Yeni $Q_{(s,a)} = (1 - \alpha) * Q_{(s,a)} + \alpha * (r_{(s,a)})$
8. $R=\text{topla}(\text{Yeni } Q_{(s,a)})$
9. If $\rightarrow R > R_{\text{önceki}}$
10. R'yi hafızaya kaydet.
11. S ortam modelini hafızaya kaydet.
12. For $\rightarrow$ Her bir dost platform için;
13. Yeni $Q_{(s,a)} = (1 - \alpha) * Q_{(s,a)} + \alpha * (r_{(s,a)})$
14. $R=\text{topla}(\text{Yeni } Q_{(s,a)})$
15. If $\rightarrow R > R_{\text{önceki}}$
16. R'yi hafızaya kaydet.
17. S ortam modelini hafızaya kaydet.