HAVA KİRLİLİĞİNİN MAKİNE ÖĞRENMESİ YÖNTEMLERİYLE TAHMİNİ

Ayça GÜVEN

T.C.
BURSA ULUDAĞ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

HAVA KİRLİLİĞİNİN MAKİNE ÖĞRENMESİ YÖNTEMLERİYLE TAHMİNİ

Ayça GÜVEN

Doç. Dr. Betül YAĞMAHAN
(Danışman)

YÜKSEK LİSANS TEZİ
ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI

BURSA – 2022

TEZ ONAYI

Ayça GÜVEN tarafından hazırlanan "Hava Kirliliğinin Makine Öğrenmesi Yöntemleriyle Tahmini" adlı tez çalışması aşağıdaki jüri tarafından oy birliği ile Bursa Uludağ Üniversitesi Fen Bilimleri Enstitüsü Endüstri Mühendisliği Anabilim Dalı'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Danışman : Doç. Dr. Betül YAĞMAHAN

Başkan	**:**	Doç. Dr. Betül YAĞMAHAN 0000-0003-1744-3062 Bursa Uludağ Üniversitesi, Mühendislik Fakültesi, Endüstri Mühendisliği Anabilim Dalı	İmza
Üye	**:**	Doç. Dr. Duygu Yılmaz EROĞLU 0000-0002-0083-4051 Bursa Uludağ Üniversitesi, Mühendislik Fakültesi, Endüstri Mühendisliği Anabilim Dalı	İmza
Üye	**:**	Doç. Dr. Aytaç YILDIZ 0000-0002-0729-633X Bursa Teknik Üniversitesi, Mühendislik ve Doğa Bilimleri Fakültesi, Endüstri Mühendisliği Anabilim Dalı	İmza

Yukarıdaki sonucu onaylarım

Prof. Dr. Hüseyin Aksel EREN
Enstitü Müdürü
../../….

B.U.Ü. Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- ve bu tezin herhangi bir bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

9/01/2022

Ayça Güven

TEZ YAYINLANMA
FİKRİ MÜLKİYET HAKLARI BEYANI

Enstitü tarafından onaylanan lisansüstü tezin/raporun tamamını veya herhangi bir kısmını, basılı (kâğıt) ve elektronik formatta arşivleme ve aşağıda verilen koşullarla kullanıma açma izni Bursa Uludağ Üniversitesi'ne aittir. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet hakları ile tezin tamamının ya da bir bölümünün gelecekteki çalışmalarda (makale, kitap, lisans ve patent vb.) kullanım hakları tarafımıza ait olacaktır. Tezde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanılması zorunlu metinlerin yazılı izin alınarak kullandığını ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederiz.

Yükseköğretim Kurulu tarafından yayınlanan "**Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge**" kapsamında, yönerge tarafından belirtilen kısıtlamalar olmadığı takdirde tezin YÖK Ulusal Tez Merkezi / B.U.Ü. Kütüphanesi Açık Erişim Sistemi ~~ve üye olunan diğer veri tabanlarının~~ (Proquest veri tabanı gibi) erişimine açılması uygundur.

ÖZET

Yüksek Lisans Tezi

Hava Kirliliğinin Makine Öğrenmesi Yöntemleriyle Tahmini

Ayça GÜVEN

Bursa Uludağ Üniversitesi
Fen Bilimleri Enstitüsü
Endüstri Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Betül YAĞMAHAN

Hava kirliliği, insanlar için dünya çapında bir risk olarak kabul edilmektedir. Uzun süre yüksek düzeyde ozon kirleticisine maruz kalmak bronşit, amfizem, astım vb. gibi kronik solunum yolu hastalıklarına yol açabilir. İnsan vücudu üzerindeki etkisine ek olarak, yüksek düzeyde ozon, mahsullerin fotosentez verimliliğini etkileyerek mahsul veriminin azalmasına da neden olur. Buna ek olarak, kentsel alanlarda hava kalitesini bozan kilit kirleticilerden biri olarak kabul edilmektedir. Bu nedenle hava kalitesinin önceden tahmin edilmesi insanları hava kirliliği konusunda uyarmak ve kontrol etmekte önemli bir rol oynamaktadır. Bu çalışmada Bursa ilindeki Bursa Uludağ Üniversitesi ve Kültürpark istasyonları için saatlik ozon O_3 hava kirleticisinin konsantrasyon değerleri makine öğrenmesi algoritmalarıyla tahmin edilmiştir. Veriler Çevre, Şehircilik ve İklim Değişikliği Bakanlığı'nın Ulusal Hava Kalitesi İzleme Ağı sitesinden elde edilmiştir. Tahminleme modeli kurulurken kirletici ve meteorolojik veriler (hava sıcaklığı, rüzgar hızı, bağıl nem ve hava basıncı) kullanılmıştır. Kullanılan makine öğrenmesi regresyon algoritmaları; rastgele orman, karar ağacı, destek vektör, k-en yakın komşu ve çok katmanlı algılayıcı regresyonudur. Regresyon algoritmalarının başarı değerleri Kök Ortalama Kare Hatası (KOKH), Ortalama Kare Hata (OKH), Ortalama Mutlak Hata (OMH), Ortalama Mutlak Yüzde Hata (OMYH) ve Açıklayıcılık Katsayısı (R^2) ile kıyaslanarak sonuçlar değerlendirilmiştir. İki istasyon için rastgele orman regresyon algoritmasının ozon konsantrasyonlarının tahmininde diğer algoritmalardan daha iyi sonuçlar verdiği görülmüştür.

Anahtar Kelimeler: Makine öğrenmesi, hava kirliliği, tahminleme
2022, ix + 59 sayfa.

ABSTRACT

MSc Thesis

Prediction of Air Pollution with Machine Learning Methods

Ayça GÜVEN

Bursa Uludağ University
Graduate School of Natural and Applied Sciences
Department of Industrial Engineering

Supervisor: Doç. Dr. Betül YAĞMAHAN

Air pollution is accepted as a worldwide risk to humans. Prolonged exposure to high levels of ozone pollutants can lead to chronic respiratory diseases such as bronchitis, emphysema, asthma, etc. In addition to its effect on the human body, high levels of ozone also affect the photosynthetic efficiency of crops, resulting in reduced crop yields. In addition, it is recognized as one of the key pollutants that degrade air quality in urban areas. Therefore, predicting air quality previously plays an important role in warning and controlling peoples about air pollution. In this study, hourly ozone air pollutant concentration values in Bursa Uludag University and Kulturpark stations for Bursa province were estimated by machine learning algorithms. The data were obtained from the National air quality monitoring network site of the Ministry of Environment, Urbanization and Climate Change. Pollutant and meteorological data (air temperature, wind speed, relative humidity and air pressure) were used in forecasting model. Random forest, decision tree, support vector, k-nearest neighbor and multilayer perceptron regression were used as the machine learning methods to forecast the O_3 values. The root-mean-square error (RMSE), mean squared error (MSE), mean absolute error (MAE), mean absolute percentage error (MAPE), and coefficient of determination (R^2) were used to evaluate the performance of the regression models. It was seen that the random forest regression algorithm for two stations gave better results in estimating ozone concentrations than other algorithms.

Key words: Machine learning, air pollution, forecasting,
2022, ix+ 59 pages.

TEŞEKKÜR

Yüksek lisans eğitimim sırasında ve tez çalışmalarım boyunca tecrübelerini benden esirgemeyen değerli danışman hocam Doç. Dr. Betül Yağmahan'a teşekkür eder, saygılarımı sunarım.
Beni bugünlere getiren anne ve babama teşekkür eder, şükranlarımı sunarım.

Ayça GÜVEN
9/01/2022

İÇİNDEKİLER

Sayfa

ÖZET i
ABSTRACT ii
TEŞEKKÜR iii
SİMGELER ve KISALTMALAR DİZİNİ vi
ŞEKİLLER DİZİNİ vii
ÇİZELGELER DİZİNİ ix
1.GİRİŞ 1
2. KURAMSAL TEMELLER ve KAYNAK ARAŞTIRMASI 3
2.1. Hava Kirleticileri 3
2.2. Makine Öğrenmesi 6
2.2.1. Denetimli öğrenme 6
2.2.2. Denetimsiz öğrenme 7
2.2.3. Pekiştirmeli öğrenme 8
2.3. Makine Öğrenmesi Algoritmaları 9
2.3.1. Destek vektör makinesi 9
2.3.2. Rastgele orman 13
2.3.3. k-en yakın komşu 15
2.3.4. Karar ağaçları 15
2.3.5. Ridge ve Lasso regresyon 17
2.3.6. Elastik net 18
2.3.7. Yapay sinir ağları 18
2.3.8. Çok katmanlı algılayıcı 19
2.4. Literatür Taraması 20
3. MATERYAL ve YÖNTEM 24
3.1. Veri Tanımı 24
3.2. Veri Ön İşleme 24
3.3. Modellerin Geliştirilmesi 24
3.3.1. Hiper parametre ayarlama 25
3.4. Modellerin Performans Değerlendirmesi 26
3.4.1. Hata metrikleri 26
3.4.1.1. Ortalama mutlak hata 26
3.4.1.2. Kök ortalama kare hata 27
3.4.1.3. Açıklayıcılık katsayısı 27
3.4.1.4. Ortalama mutlak yüzde hata 27
3.4.1.5. Ortalama kare hata 27
4. BULGULAR ve TARTIŞMA 28
4.1. Bursa Uludağ Üniversitesi İstasyon Sonuçları 28
4.1.1. Karar ağacı regresyon sonuçları 28
4.1.2. Rastgele orman regresyon sonuçları 30
4.1.3. Destek vektör makinesi regresyon sonuçları 32
4.1.4. Çok katmanlı algılayıcı regresyon sonuçları 33
4.1.5. k-en yakın komşu regresyon sonuçları 35
4.2. Kültürpark İstasyon Sonuçları 36
4.2.1. Karar ağacı regresyon sonuçları 37
4.2.2. Rastgele orman regresyon sonuçları 38
4.2.3. k-en yakın komşu regresyon sonuçları 40

4.2.4. Çok katmanlı algılayıcı regresyon sonuçları 41
4.2.5. Destek vektör makinesi regresyon sonuçları 43
5. SONUÇ 45
KAYNAKLAR 46
EKLER 51
ÖZGEÇMİŞ 59

SİMGELER ve KISALTMALAR DİZİNİ

Simgeler	**Açıklama**
C	Ceza parametresi
ε	Sapma
ε_i	Gevşek değişken
k	Sınıf merkezi

Kısaltmalar	**Açıklama**
OMH	Ortalama mutlak hata (MAE-Mean Absolute Error)
OKH	Ortalama kare hata (MSE-Mean Squared Error)
OMYH	Ortalama mutlak yüzde hata (MAPE-Mean Absolute Percentage Error)
KOKH	Kök ortalama kare hatası (RMSE-Root Mean Square Error)
R^2	Açıklayıcılık katsayısı (R-Squared)
MÖ	Makine öğrenmesi (ML-Machine Learning)
YZ	Yapay zeka (AI-Artificial Intelligent)
PM	Partikül madde
SO_2	Kükürt dioksit
NO_2	Azot dioksit
O_3	Ozon
CO_2	Karbondioksit
CO	Karbon monoksit
SOX	Sülfür oksit
NOX	Azot oksit
VOC	Uçucu organik bileşikler
DR	Doğrusal Regresyon (LR-Linear Regression)
DVM	Destek vektör makinesi (SVM-Support Vector Machine)
DVR	Destek vektör regresyon (SVR-Support Vector Regression)
k-EK	k-en yakın komşu (kNN-k-Nearest Neighbour)
KA	Karar ağacı (DT-Decision Tree)
RR	Ridge regresyon (Ridge Regression)
YSA	Yapay sinir ağı (ANN-Artificial Neural Network)
ÇKA	Çok katmanlı algılayıcı (MLP-Multilayer Perceptron)
RO	Rastgele Orman (RF-Random Forest)
CART	Sınıflandırma ve regresyon ağaçları (Classification and Regression Trees)
XGBoost	Ekstrem gradyan arttırma (Extreme Gradient Boost)

ŞEKİLLER DİZİNİ

Sayfa

Şekil 2.1. Denetimli öğrenme modeli 7
Şekil 2.2. Denetimsiz öğrenme modeli 8
Şekil 2.3. Sınıflandırma için DVM 9
Şekil 2.4. Regresyon için DVM 12
Şekil 2.5. RO regresyonu 14
Şekil 2.6. KA yapısı 16
Şekil 2.7. YSA yapısı 19
Şekil 4.1. Bursa Uludağ Üniversitesi istasyonu için 1-12 Kasım tarihleri arasında KA regresyon kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerinin karşılaştırma grafiği 29
Şekil 4.2. Bursa Uludağ Üniversitesi istasyonu için KA regresyon kullanarak tahmin edilen saatlik O_3 değerlerinin dağılım grafiği 30
Şekil 4.3. Bursa Uludağ Üniversitesi istasyonu için 1-12 Kasım tarihleri arasında RO regresyon kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerinin karşılaştırma grafiği 31
Şekil 4.4. Bursa Uludağ Üniversitesi istasyonu için RO regresyon kullanarak tahmin edilen saatlik O_3 değerlerinin dağılım grafiği 32
Şekil 4.5. Bursa Uludağ Üniversitesi istasyonu için 1-12 Kasım tarihleri arasında DVR kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerinin karşılaştırma grafiği 33
Şekil 4.6. Bursa Uludağ Üniversitesi istasyonu için DVR kullanarak tahmin edilen saatlik O_3 değerlerin dağılım grafiği 33
Şekil 4.7. Bursa Uludağ Üniversitesi istasyonu için 1 -12 Kasım tarihleri arasında ÇKA regresyon kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerinin karşılaştırma grafiği 34
Şekil 4.8. Bursa Uludağ Üniversitesi istasyonu için ÇKA regresyon kullanarak tahmin edilen saatlik O_3 değerlerinin dağılım grafiği 35
Şekil 4.9. Bursa Uludağ Üniversitesi istasyonu için 1-12 Kasım tarihleri arasında k-EK regresyon kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerinin karşılaştırma grafiği 36
Şekil 4.10. Bursa Uludağ Üniversitesi istasyonu için k-EK regresyon kullanarak tahmin edilen saatlik O_3 değerlerinin dağılım grafiği 36
Şekil 4.11. Kültürpark istasyonu için 1-12 Kasım tarihleri arasında KA regresyon kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerinin karşılaştırma grafiği 38
Şekil 4.12. Kültürpark istasyonu için KA regresyon kullanarak tahmin edilen saatlik O_3 değerlerinin dağılım grafiği 38
Şekil 4.13. Kültürpark istasyonu için 1-12 Kasım tarihleri arasında RO regresyon kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerinin karşılaştırma grafiği 39
Şekil 4.14. Kültürpark istasyonu için RO regresyon kullanarak tahmin edilen saatlik O_3 değerlerinin dağılım grafiği 40
Şekil 4.15. Kültürpark istasyonu için 1-12 Kasım tarihleri arasında k-EK regresyon kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerinin karşılaştırma grafiği 41

Şekil 4.16. Kültürpark istasyonu için k-EK regresyon kullanarak tahmin edilen saatlik O_3 değerlerinin dağılım grafiği.....41
Şekil 4.17. Kültürpark istasyonu için 1-12 Kasım tarihleri arasında ÇKA regresyon kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerinin karşılaştırma grafiği 42
Şekil 4.18. Kültürpark istasyonu için ÇKA regresyon kullanarak tahmin edilen saatlik O_3 değerlerinin dağılım grafiği.....43
Şekil 4.19. Kültürpark istasyonu için 1-12 Kasım tarihleri arasında DVR kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerinin karşılaştırma grafiği 44
Şekil 4.20. Kültürpark istasyonu için DVR kullanarak tahmin edilen saatlik O_3 değerlerinin dağılım grafiği.....44

ÇİZELGELER DİZİNİ

Sayfa

Çizelge 4.1. Bursa Uludağ Üniversitesi istasyonu için O_3 tahmininde kullanılan makine öğrenmesi algoritmalarının sonuçları 28
Çizelge 4.2. Bursa Uludağ Üniversitesi istasyonu için KA regresyonda GridSearchCV kullanarak en iyi parametreleri ayarlama 29
Çizelge 4.3. Bursa Uludağ Üniversitesi istasyonu için RO regresyonda GridSearchCV kullanarak en iyi parametreleri ayarlama 31
Çizelge 4.4. Bursa Uludağ Üniversitesi istasyonu için DVR algoritmasında GridSearchCV kullanarak en iyi parametreleri ayarlama 32
Çizelge 4.5. Bursa Uludağ Üniversitesi istasyonu için ÇKA regresyonda GridSearchCV kullanarak en iyi parametreleri ayarlama 34
Çizelge 4.6. Bursa Uludağ Üniversitesi istasyonu için k-EK regresyonda GridSearchCV kullanarak en iyi parametreleri ayarlama 35
Çizelge 4.7. Kültürpark istasyonu için O_3 tahmininde kullanılan makine öğrenmesi algoritmalarının sonuçları 37
Çizelge 4.8. Kültürpark istasyonu için KA regresyonda GridSearchCV kullanarak en iyi parametreleri ayarlama 37
Çizelge 4.9. Kültürpark istasyonu için RO regresyonda GridSearchCV kullanarak en iyi parametreleri ayarlama 39
Çizelge 4.10. Kültürpark istasyonu için DVR algoritmasında GridSearchCV kullanarak en iyi parametreleri ayarlama 40
Çizelge 4.11. Kültürpark istasyonu için ÇKA regresyonda GridSearchCV kullanarak en iyi parametreleri ayarlama 42
Çizelge 4.12. Kültürpark istasyonu için DVR algoritmasında GridSearchCV kullanarak en iyi parametreleri ayarlama 43

1.GİRİŞ

Hava, dünyadaki tüm yaşam için gerekli olan oksijen ve diğer gazları sağladığı için çok önemlidir ve tüm canlıların hayatta kalması için hayati bir unsurdur; bu nedenle, temiz ve güvenli tutmak gereklidir. Hava kirliliğinin başlıca nedenleri arasında ekonomik gelişme, kentleşme, enerji tüketimi, ulaşım ile kent nüfusunun hızla artması yer almaktadır. Günlük hayatımızda karşılaştığımız en büyük hava kirleticileri partikül madde (PM), kükürt dioksit (SO_2), azot dioksit (NO_2), ozon (O_3), karbon monoksit (CO) ve karbondioksittir (CO_2) (Bozdağ, Dokuz ve Gökçek, 2020).

O_3, yüksek reaktiviteye sahip renksiz ve kokusuz bir gazdır ve doğrudan havaya yayılmadığı ve atmosferdeki karmaşık kimyasal reaksiyonlardan kaynaklandığı için kirleticiler arasında benzersizdir. Ozon, atmosferin iki bölümü olan stratosferde (yeryüzünden 20-30 km arasındaki katman) ve troposferde (yer seviyesinden 15 km'ye kadar olan katman) eser miktarda oluşan reaktif bir oksidandır. "Ozon tabakası" olarak da bilinen stratosferik ozon, doğal olarak oluşur ve güneşin biyolojik olarak zararlı ultraviyole radyasyonunun bir kısmını emerek koruyucu bir kalkan oluşturduğundan, insanlar ve diğer yaşam türleri için faydalı olduğu düşünülmektedir (Ben Ishak ve ark. 2017). Yer seviyesindeki ozon, akciğer dokusuna, bitkilere ve diğer canlı sistemlere zarar veren ve doğrudan havaya yayılmayan, uçucu organik bileşikler (VOC) ve azot oksitler (NOX, NO ve NO_2 kombinasyonu) arasındaki güneş ışığı ve ısı varlığında kimyasal reaksiyonla oluşan zararlı bir kirleticidir. Yer seviyesindeki O_3, organik bileşikler, motorlu taşıtlar ve diğer endüstriyel kaynaklar dahil olmak üzere çeşitli kaynaklardan yayılır ve özellikle sıcak güneşli kentsel alanlarda atmosferde kolayca oluşur. O_3, bulunduğu yere bağlı olarak önemli ölçüde farklı etkilere sahiptir; dünyadaki yaşama zarar verebilir veya yaşamı koruyabilir. Tahmin edilmesi ve kontrol edilmesi zor olan ciddi bir çevre sorunu olmasının ana nedeni budur (Rajab ve diğerleri, 2013). Yüksek ozon seviyeleri, akciğer hastalıkları dahil olmak üzere solunum sağlığı sorunlarına ve erken ölümlere neden olabilir. Bu nedenle hava kirliliği kontrolü kaçınılmazdır ve hava kalitesinin doğru tahmin edilmesi hava kalitesi yönetiminin en önemli parçasıdır. Ancak, karmaşık fiziksel ve kimyasal süreçleri nedeniyle hava kalitesini doğru bir şekilde tahmin etmek zor bir iştir. Meteorolojik ve kirlilik verileri gibi doğrusal olmayan zaman serisi bilgilerini tahmin etmek için makine öğrenimi yöntemlerine son zamanlarda artan bir ilgi

vardır. Makine Öğrenmesi (MÖ), özellikle denetimli, denetimsiz, vb. gibi farklı yollarla hava kirliliği ile ilgili tahmin ve optimizasyonda en umut verici yöntemdir. Geleneksel hava kalitesi tahmin yöntemlerinin kirletici konsantrasyonunun tahmini için daha fazla hesaplama gücü gerektirdiği göz önünde bulundurulduğunda, birçok araştırmacı daha iyi sonuçlara yol açabilecek Yapay Zeka (YZ) algoritmalarını (makine öğrenimi, derin öğrenme vb.) uygulamaya çalışmaktadır. Makine öğrenimi modelleri, çevresel çalışmalarla ilgili olanlar da dahil olmak üzere çok çeşitli uygulamalarda iyi performans göstermektedir (Yafouz ve diğerleri, 2021).

Bu çalışmanın amacı, Bursa ilindeki Uludağ Üniversitesi ve Kültürpark istasyonları için makine öğrenmesi yöntemleri kullanarak saatlik ozon (O_3) konsantrasyonlarını tahmin etmektir. Kullanılan makine öğrenmesi regresyon algoritmaları; rastgele orman (RO), karar ağacı (KA), destek vektör makinesi (DVM), k-en yakın komşu (k-EK) ve çok katmanlı algılayıcı (ÇKA) regresyondur.

Regresyon modellerinin performansını değerlendirmek için Kök Ortalama Kare Hatası (KOKH), Ortalama Kare Hata (OKH), Ortalama Mutlak Hata (OMH), Ortalama Mutlak Yüzde Hata (OMYH) ve Açıklayıcılık Katsayısı (R^2) kullanılmıştır.

2. KURAMSAL TEMELLER ve KAYNAK ARAŞTIRMASI

Hava kirliliği, belirli gazların ve partiküllerin atmosferde sağlığımıza zarar verebilecek, solunum problemlerine neden olabilecek ve hatta erken ölüme yol açabilecek, çevreye zarar verebilecek düzeyde biriktiğinde ortaya çıkar (Yafouz ve diğerleri, 2021). Bu gazlar ve partüküller kirleticiler olarak bilinirler. Kirleticiler; kömür, petrol veya dizel gibi fosil yakıtların yakılması da dahil olmak üzere insan yapımı kaynaklardan veya volkanik patlamalar ve orman yangınları gibi doğal kaynaklardan oluşabilirler. Hava kirleticileri, birincil ve ikincil kirletici olarak sınıflandırılır ve katı parçacık, sıvı damlacık veya gaz şeklinde olabilir (Suárez Sánchez ve diğerleri, 2011).

Kaynaktan doğrudan atmosfere yayılan kirleticiler, birincil kirleticilerdir. Kaynaklar, kum fırtınaları gibi doğal süreçler veya endüstri ve araç emisyonları gibi insan yapımıyla olabilir. En yaygın birincil kirleticiler kükürt dioksit (SO_2), partikül madde (PM), nitrojen dioksit (NO_2) ve karbon monoksittir (CO) (Suárez Sánchez ve diğerleri, 2011).

Birincil kirleticiler arasındaki kimyasal veya fiziksel etkileşimlerden kaynaklanarak atmosferde oluşan hava kirleticileri ikincil kirleticilerdir. İkincil kirleticiler doğrudan havaya yayılmaz. İkincil kirleticiler üreten birçok reaksiyon, güçlü güneş ışığı tarafından tetiklenir ve bu nedenle fotokimyasal reaksiyonlar olarak adlandırılır. Fotokimyasal oksitleyiciler, ikincil partikül madde, ozon ikincil kirleticilerin başlıca örnekleridir (Castelli ve diğerleri, 2020).

2.1. Hava Kirleticileri

Karbon monoksit (CO) renksiz, kokusuz ve çok zehirli bir gazdır. Petrol, kömür veya odun yakma gibi karbon bazlı enerji kaynaklarını içeren tamamlanmamış yanma işlemlerinin bir yan ürünü olarak oluşur. İşlem sırasında, yan ürün olarak karbondioksit oluşturmak için yeterli oksijen yoksa, bunun yerine karbon monoksit oluşur. En büyük karbon monoksit kaynakları, özellikle yavaş hareket ettiklerinde veya motor rölantideyken araçlardan gelir. Karbon monoksit insanlar için tehlikelidir, solunduğunda kırmızı kan hücrelerindeki hemoglobine bağlanarak oksijenle rekabet eder ve beyin, sinir sistemi dokuları ve oksijen kalbi gibi hayati organları aç bırakır ve düzgün çalışma yeteneklerini azaltır (SEPA, 2022).

Azot oksitler (NOX), değişen miktarlarda oksijen ve azot moleküllerinden oluşan bir gaz grubudur. Enerji santralleri ve motorlu taşıtlar birincil kaynaklardır. Yakıttaki veya havadaki azotun oksijenle reaksiyona girmesiyle yakıtın yüksek sıcaklıkta yanması sırasında oluşan gazlardır. Bu gazlar ayrıca belirli bakteriler azot içeren bileşikleri oksitlediğinde doğal olarak oluşur. Oluşan ilk ürün nitrik oksittir (NO). NO atmosferde daha fazla oksitlendiğinde, nitrojen dioksit (NO_2) oluşur. En yaygın nitrojen oksitlerden biri, hoş olmayan bir kokusu olan ve yüksek konsantrasyonlarda zehirli olan kırmızımsı, kahverengi bir gaz olan nitrojen dioksittir (NO_2). Bunlar ikincil kirleticiler oluşturabilir ve asitlenme ve azot zenginleştirme gibi çevresel sorunlara neden olabilir. Fosil yakıtlar yüksek sıcaklıklarda yakıldığında oluşurlar, ancak yıldırım çarpmalarıyla doğal olarak da oluşabilirler. Kentsel alanlardaki nitrojen dioksitin çoğu egzoz emisyonlarından gelir. Akciğerlerin astarını alevlendirdiği için solunum problemleri olasılığını artırabilir ve akciğer enfeksiyonlarına karşı bağışıklığı azaltabilir. Bu hırıltı, öksürük, soğuk algınlığı, grip ve bronşit gibi sorunlara neden olabilir (SEPA, 2022).

Sülfür oksitler (SOX), kükürt ve oksijen moleküllerinden oluşan bir grup bileşiktir. En yaygın kükürt oksit, yanık kibrit kokusuna sahip renksiz bir gaz olan kükürt dioksittir (SO_2). Bunlar ikincil kirleticiler oluşturabilir ve asitlenme gibi çevresel sorunlara neden olabilir. Kömür ve yağ gibi kükürt içeren yakıtların ve metal içeren cevherlerin (alüminyum, bakır, çinko, kurşun ve demir dahil) yanması sırasında oluşurlar. Havamızdaki kükürt dioksitin çoğu, enerji üretimi ve endüstriyel faaliyetler için kömür ve petrolün yakılmasından kaynaklanmaktadır. Ayrıca aktif volkanlar ve kaplıcalar gibi doğal kaynaklardan da oluşmaktadır. Kükürt dioksit vücuda solunması halinde solunum güçlüklerine neden olabilir. Ayrıca bitkiler için zehirlidir ve havadaki nemle reaksiyona girdiğinde asit yağmurlarına neden olabilir (SEPA, 2022).

Partiküller veya partikül madde (PM), havadaki küçük katı madde veya sıvı parçalarıdır ve karbon, kükürt, nitrojen ve metal bileşikleri dahil yüzlerce farklı kimyasaldan oluşabilir. Bazıları çıplak gözle görülebilecek kadar büyüktür, bazıları ise sadece güçlü mikroskoplarla görülebilir. Daha büyük partiküller, PM_{10}(10 mikrometreden büyük) genellikle burun ve boğaz yoluyla vücuttan süzülür. 10 mikrometre veya daha küçük partiküller, akciğerlerin en derin kısımlarına solunabilir. İnce partiküller, $PM_{2.5}$ (2,5

mikrometreden küçük) akciğerlerden kan dolaşımına geçecek kadar küçüktür. İnce partiküller motorlu taşıtlar, elektrik üretimi ve endüstriyel tesislerin yanı sıra konut şömineleri ve odun sobalarından kaynaklanan yakıt yanmasından kaynaklanır (Suárez Sánchez ve ark. 2011). Partikül maddelere kısa süreli maruz kalma akciğer hastalıklarına neden olur ve düşük konsantrasyonlara uzun süreli maruz kalma kanser ve bebek ölümlerine neden olur (Bozdağ, Dokuz ve Gökçek, 2020).

Rüzgar, alçak basınç ve yüksek basınç bölgesi arasında yer değiştiren ve daima yüksek basınç bölgesinden alçak basınç bölgesine doğru hareket eden hava akımıdır. İki bölge arasındaki basınç farkı ne kadar büyükse, hava akış hızı da o kadar büyük olur. Normal hava şartlarında güneşin dünyayı ısıtması sonucu yerkürenin üzerindeki hava tabakası ısınır. Dünyanın hemen üzerindeki ısınan hava tabakası (topraktan uzaklaştıkça troposfer tabakasının sıcaklığı düştüğü için) yukarı doğru yükselir. Yeryüzünde kirli bir hava tabakası varsa bu olay sonucunda kirlenen hava tabakası doğal olarak dünyadan uzaklaşır. Atmosfer inversiyonunda; özellikle sonbahar ve kış aylarında yerkürenin hızlı soğuması sonucu yerküre üzerindeki hava tabakası soğur. Bu soğutucu hava tabakası üst sıcak hava tabakasını geçemediği için toprak ile sıcak hava tabakası arasında hapsolur. Bu hapsolmuş hava tabakasında biriken kirleticiler, yerin hemen üzerinde kirli bir hava tabakasının oluşmasına neden olur. Atmosferik inversiyon sonucunda şehrin üzerinde hapsolan kirli hava tabakası buradaki canlıları olumsuz etkiler. Bu kirli hava tabakası ancak çok kuvvetli hava akımları (rüzgarlar) sonucu dağılabilir (Şahin, Işık, Şahin ve Kara, 2020).

Ozon (O_3) moleküler oksijenin (O_2) üç atomlu bir formudur. Madde keskin kokulu, zehirli, soluk mavi, kararsız bir gazdır. (O_3), birincil kirleticilerin kimyasal reaksiyonu sonucu oluşan ikincil bir kirleticidir. Ozon, özellikle stratosferde dünya yüzeyinden 19 ile 30 km uzaklıktadır. Bu yüksekliklerde ozon, yeryüzüne inen ultraviyole (UV) radyasyonu filtreler. Dünya düzeyinde ozon, insan sağlığı için önemli bir tehdit oluşturmaktadır. Ozon güçlü bir oksitleyicidir (Maleki ve diğerleri, 2019; Şahin ve diğerleri, 2020). Yüksek ozon seviyelerine kısa süreli maruz kalma, göz ve akciğer tahrişlerine neden olur. Ayrıca, daha orta düzeylerde uzun süreli veya tekrarlayan maruziyetlerin kronik etkilerine dair artan kanıtlar vardır. Ozon oluşturan reaksiyonlar güçlü güneş ışığı ile uyarıldığından, bu kirleticinin oluşumu gündüz saatleri ile sınırlıdır.

Zirveler, bir dizi sıcak güneşli ve sakin günün ardından öğleden sonra meydana gelir. Tahmin edebileceğimiz gibi, ozon seviyeleri daha sıcak yaz aylarında en yüksek seviyededir (Suárez Sánchez ve diğerleri, 2011).

2.2. Makine Öğrenmesi

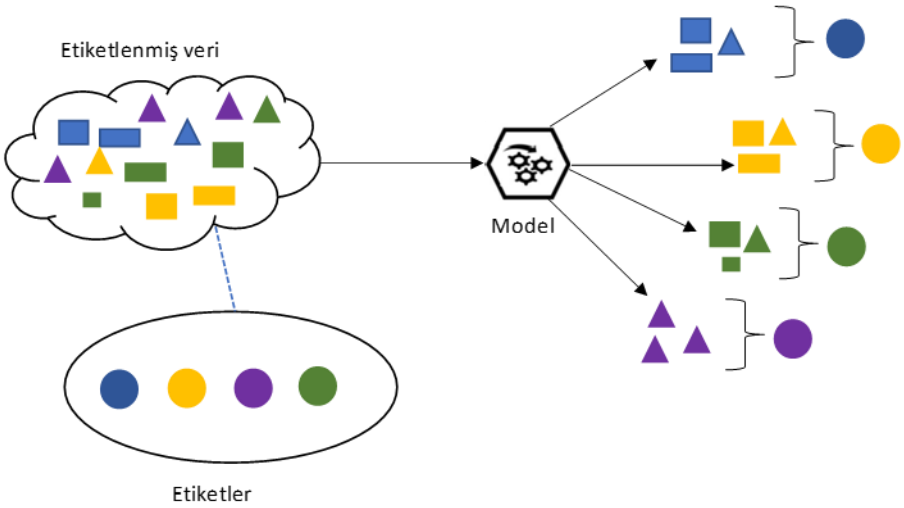
Makine öğrenimi, bilgisayarların sorunlara kendi başlarına çözüm bulmayı öğrenmelerini sağlayan bilgisayar biliminin dalıdır. Makine öğrenimi, verilere erişebilen ve bunları kendileri için öğrenmek için kullanabilen bilgisayar programlarının geliştirilmesine odaklanır. Başka bir deyişle, makine öğrenimi, bilgisayarın açıkça programlanmadan çözümler bulmasını sağlar (Pattnaik ve diğerleri, 2020). Genel bir bilgisayar programı girdiyi almayı, verilen talimatlar üzerinden işlemeyi ve çıktıyı vermeyi amaçlar. Makine öğrenimi ise çözüme götüren en uygun algoritmayı bulmak için sağlanan girdiye ve sorunun çözümüne odaklanır. Makine öğrenimi, sistemlere açıkça programlanmadan deneyimlerden otomatik olarak öğrenme ve iyileştirme yeteneği sağlayan bir yapay zeka uygulamasıdır (Guabassi ve diğerleri, 2021). Yapay zeka uygulamalarından farklı olarak, makine öğrenimi, veri içindeki gizli kalıpların öğrenilmesini (veri madenciliği) ve ardından problemle ilgili bir olayı sınıflandırmak veya tahmin etmek için kalıpları kullanmayı içerir. Tüm yapay zeka yöntemleri makine öğrenmesi algoritmaları olarak nitelendirilmese de, tüm makine öğrenmesi algoritmalarının da yapay zeka teknikleri olduğunu belirtmek yeterlidir.

Makine öğrenimi, denetimli öğrenme, denetimsiz öğrenme ve pekiştirmeli öğrenme olmak üzere üç grupta incelenebilir.

2.2.1. Denetimli öğrenme

Denetimli makine öğrenimi, girdi özniteliklerinin yanı sıra önceden belirlenmiş çıktı özniteliğini de içerir. Algoritmalar, önceden belirlenmiş özniteliği tahmin etmeye ve sınıflandırmaya çalışır. En yaygın denetimli öğrenme yöntemleri regresyon ve sınıflandırmadır. Regresyon, test puanları, laboratuvar değerleri veya bir öğenin fiyatları gibi sayısal verilerin tahmin edilmesini içerir. Sınıflandırma ise bir örneğin hangi kategoriye ait olduğunu tahmin etmeyi gerektirir. Çıktı niteliksel olduğunda, makine öğrenimi sınıflandırma görevini yerine getirir (Vakharia ve diğerleri, 2021). Şekil 2.1, denetimli modelin bir örneğini göstermektedir. Örnek olarak verilen model, renklerine

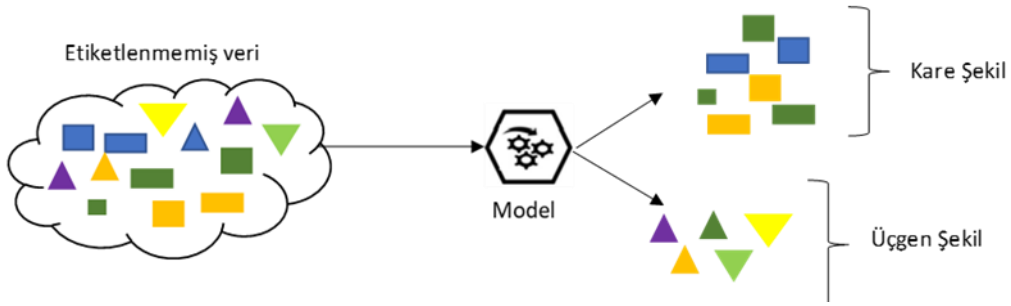
göre etiketlenmiş geometrik şekillerin sınıflandırılmasını sağlar (Taoufik ve diğerleri, 2021).



Şekil 2.1. Denetimli öğrenme modeli (Taoufik ve diğerleri, 2021)

2.2.2. Denetimsiz öğrenme

Denetimsiz öğrenme, bir hedef özniteliğin katılımı olmadan örüntü tanımayı içerir. Yani analizde kullanılan tüm değişkenler girdi olarak kullanılır. En yaygın denetimsiz öğrenme yöntemlerinden bazıları kümeleme, ilişkilendirme ve anormallik algılamadır. Bu yöntemlerde, bir veri kümesinde var olan veya olmayan örüntüler bir hedef tarafından bilgilendirilmez ve algoritma tarafından belirlenmeye bırakılır. Denetimsiz öğrenme, modelin daha önce tespit edilmemiş kalıpları ve bilgileri keşfetmek için kendi başına çalışmasına izin veren bir makine öğrenimi tekniğidir. Şekil 2.2'de denetimsiz model verileri geometrik şekle göre tahmin etmiştir. Sonuç tatmin edici değilse, verilerde başka kalıplar bulmak için yeniden eğitilmesi istenir. Örneğin, alternatif bir sınıflandırma, yüzeye veya renge göre olabilir. Doğrulayarak renk seçimiyle, model her yeni şekil için rengi (sarı, turuncu, mor ve yeşil) belirtebilir (Taoufik ve diğerleri, 2021).



Şekil 2.2. Denetimsiz öğrenme modeli (Taoufik ve diğerleri, 2021)

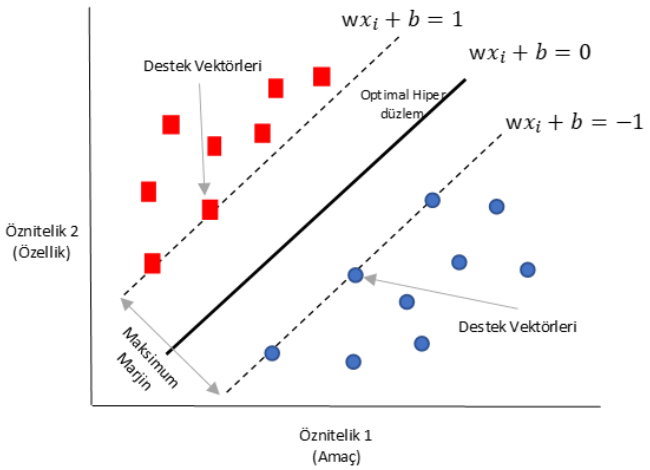
2.2.3. Pekiştirmeli öğrenme

Takviyeli öğrenme, deneyime dayalı öğrenmeyi ifade eder, yani bir sistem bir çevre ile etkileşime girer ve eylemlerinin sonucuna göre öğrenir. Takviyeli öğrenme uygulamasına bir örnek, simülasyonlar aracılığıyla sondaj sırasında çamur ağırlığının (çamur yoğunluğunun) nasıl ayarlanacağını öğrenen bir robottur. Robot, farklı sondaj senaryolarında kullanmak üzere çamur ağırlığının doğru ve yanlış değerlerine karar vermede yaptığı seçimlerin sonucunu kullanabilir (Osarogiagbon ve diğerleri, 2021). Ajan olarak adlandırılan öğrenme sistemi, bir dizi eylem yoluyla çevresini kendi başına bilmeyi öğrenir. Olumlu eylemin gerçekleştirilmesinde, ajana bir ödül (olumlu puan) ve tam tersi durumda bir ceza (negatif puan) verilir. Böylece, ödülleri maksimize ederek ve cezalardan kaçınarak, ajan, kendi başına, çevresiyle ilgili olarak benimsemesi gereken en iyi stratejiyi öğrenir. Başka bir deyişle, bu yöntem, ajanın insan müdahalesi olmadan ve ilgili görevin açık bir şekilde programlanması olmadan son puanı en üst düzeye çıkarmak için bir dizi karar vermesine izin verir (Taoufik ve diğerleri, 2021).

2.3. Makine Öğrenmesi Algoritmaları

2.3.1. Destek vektör makinesi

DVM ilk olarak sınıflandırma problemleri için 1995 yılında Vapnik tarafından geliştirilmiştir (Cortes ve Vapnik, 1995). DVM'nin sınıflandırma versiyonu Şekil 2.3'te gösterilmiştir. Sınıfların sınırları üzerinde yer alan noktalara destek vektörleri, aradaki uzaya ise hiperdüzlem adı verilir. Bu iki bölge arasındaki boşluk, sınıflar arasındaki marjindir. Hiper düzlemler, veri setinde meydana gelen sınıfların sayısını belirler ve görünmeyen verilerin çıktısı, yeni verilerle hangi sınıfın en fazla benzerliğe sahip olduğuna göre tahmin edilir (Liang ve diğerleri, 2020). Lineer bir ayırıcı bir çözüm bulamadığı zaman, veri noktaları daha yüksek boyutlu bir uzaya yansıtılır, burada önceki lineer olmayan ayrılabilir noktalar kernel fonksiyonları kullanılarak lineer olarak ayrılabilir hale gelir. DVM, denetimli öğrenme yöntemleri alanına aittir ve bu nedenle yeni görünmeyen verileri sınıflandırmak için etiketlenmiş, bilinen verilere ihtiyaç duyar. Verileri sınıflandırmak için temel yaklaşım, veri noktalarını mümkün olan en az hata miktarıyla veya mümkün olan en büyük marjla ilgili etiketlere bölen bir fonksiyon yaratmaya çalışmakla başlar. DVM, destek vektörleri ve hiper düzlem arasındaki marjı maksimize etmeyi ve sınıflar arasında en uygun ayırıcı hiper düzlemi bulmayı amaçlar (Taoufik ve diğerleri, 2021).



Şekil 2.3. Sınıflandırma için DVM (Liang ve diğerleri, 2020)

Optimum ayırıcı hiper düzlem, sınıfların vektörleri arasında maksimum marjı olan bir karar fonksiyonudur. Marj, optimal hiper düzlemi oluşturmak için kullanılan "destek vektörleri" adı verilen eğitim verilerinin küçük kısmı tarafından belirlenir. Eğitim seti boyutuna göre az sayıda destek vektörü ile optimal hiper düzlem oluşturulabilirse, DVM'nin genelleme yeteneğinin yüksek olacağı gösterilmiştir (Cortes ve Vapnik, 1995).

Her x_i girdisinin y_i = –1 veya +1 iki sınıftan birinde olduğu, l eğitim noktalarına sahip iki sınıflı bir sınıflandırma problemi için, (x_1 , y_1), …, (x_l , y_l) eğitim verileri verildiğini varsayalım. $x_i \in R^n$ i = 1,2, . . . , l ve herbir girdi vektör x_i için hedef vektör $y_i \in (-1, +1)$ olur.

Optimal hiper düzlem $wx_i + b = 0$ olarak tanımlanır. Burada w ağırlık vektörünü, b eğilim değerini göstermektedir. w ve b, eğitim setinin tüm elemanları için aşağıdaki eşitsizlikleri sağlar.

$$wx_i + b \geq +1, y_i = 1 \tag{2.1}$$

$$wx_i + b \leq -1, y_i = -1 \tag{2.2}$$

Bir DVM modelini eğitmenin amacı, hiper düzlemin verileri ayırması ve marjı $\frac{1}{\|w\|^2}$ maksimize etmesi için w ve b'yi bulmaktır. $y_i(wx_i + b) = 1$ destek vektörler olarak adlandırılır.

Optimum ayırma hiper düzlemi, aşağıdaki optimizasyon problemi çözülerek belirlenebilir (Cortes ve Vapnik, 1995).

$$min\ \frac{1}{2}w^2 + C\sum_{i=1}^{l}\xi_i \tag{2.3}$$

$$y_i(w\varphi(x_i) + b)1 - \xi_i\ ,\ \xi_i \geq 0 \tag{2.4}$$

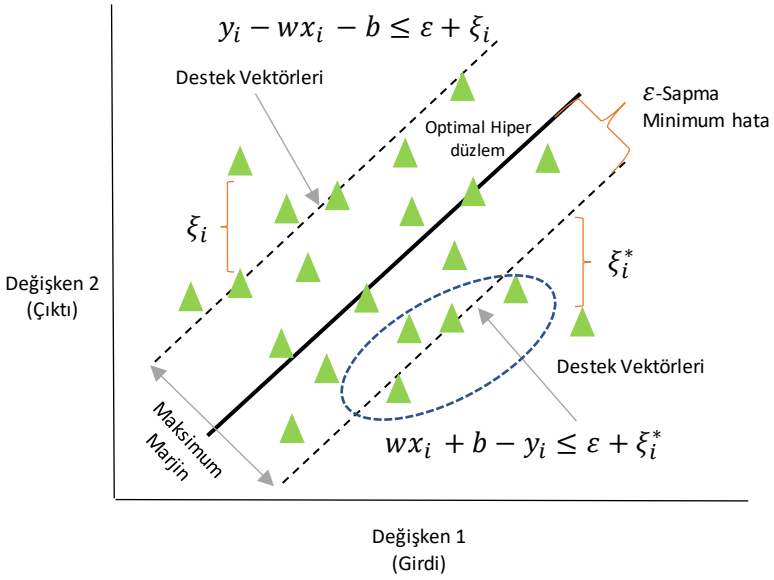
w vektörü ve b skaleri, ayırma hiper düzleminin optimal yönelimini tanımlar. Gevşek değişkenler ξ_i, ayrılamayan eğitim verilerine izin vermek için kullanılır. C ceza parametresidir ve marjı maksimize etmenin ve bolluğu minimuma indirmenin göreli önemini belirler. Doğrusal olarak ayrılamayan veriler için, x girdi vektörünü daha yüksek boyutlu bir özellik uzayına eşlemek için çekirdek φ aşağıdaki gibi kullanılır:

$$K(x, x_i) = \varphi(x).\varphi(x_i) \tag{2.5}$$

$K(x, x_i)$, Mercer koşulunu sağlayan olası karar verilmiş bir fonksiyondur (Vapnik, 2000). Girdi uzayında farklı tipte doğrusal olmayan karar yüzeylerine sahip öğrenen makineler, farklı çekirdekler kullanılarak oluşturulabilir. Polinom öğrenme makineleri, radyal tabanlı fonksiyon makineleri ve sigmoid (iki katmanlı sinir ağları), doğrusal olmayan karar yüzeyleri ile en çok çalışılan öğrenme makineleri arasındadır. Radyal tabanlı fonksiyon (Gauss) aşağıdaki biçimdedir:

$$K(x, x_i) = exp\left(-\frac{|x-x_i|^2}{2\sigma^2}\right) \tag{2.6}$$

DVM'nin regresyon versiyonu Şekil 2.4'te gösterilmiştir. Bir hiper düzlemin lineer olmayan bir fonksiyona yaklaşımı, lineer regresyon ile maksimum marjinde inşa edilmiştir. Bu nedenle, ε bölgesi içinde bulunan bazı sapmaları tolere etmek için ε duyarsız kayıp olarak bilinen ek parametre tanıtılır. DVR modelindeki hiper düzlem (düz çizgi) boyunca sınır çizgileri (kesik çizgiler) ε parametresine göre tanımlanır, burada ortaya çıkan çizgiler hiper düzlemden $-\varepsilon$ ve $+\varepsilon$ miktarındaki kaydırılmış fonksiyondur. SVR, yukarıdaki (ξ_i) veya altındaki (ξ_i^*) sınırların dışındaki çıktı değişkenleri için C parametresi (maliyet faktörü) tarafından sunulan bir ceza kullanır. Bununla birlikte, sınırlar içindeki veri noktaları muaf tutulur. Destek vektörleri bu sınır çizgilerinin yakınında bulunan veri noktalarını temsil ettiğinden, ε hiper düzlemden daha uzağa hareket ederse destek vektörlerinin sayısı azalır; aksi halde, ε hiper düzleme yaklaştıkça destek vektörlerinin sayısı artar (Liang ve diğerleri, 2020).



Şekil 2.4. Regresyon için DVM (Liang ve diğerleri, 2020)

Birincil DVR sorunu aşağıdaki şekilde tanımlanabilir (Smola ve Schölkopf, 2004):

$$\min \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{l} (\xi_i + \hat{\xi}_i) \tag{2.7}$$

$$wx_i + b - y_i \leq \varepsilon + \xi_i \quad i = 1, \ldots, l, \tag{2.8}$$

$$y_i - wx_i - b < \varepsilon + \xi_i \quad i = 1, \ldots, l, \tag{2.9}$$

$$\xi_i, \hat{\xi}_i \geq 0, \qquad i = 1, \ldots, l, \tag{2.10}$$

w, d boyutlu bir ağırlık vektörüdür. $C > 0$ sabiti, ε 'dan büyük sapma üst sınırının hala tolere edilebildiği, karar fonksiyonundaki farklılıklar arasındaki dengeyi belirler.
ε' dan büyük bir sapma C cezasına tabi olacaktır. Ayrıca, yüksek gevşek değişken değerleri, deneysel hataların düzenleyici faktörleri önemli ölçüde etkilemesine neden olur. DVR'de destek vektörü, karar fonksiyonunun sınırları üzerinde veya dışında bulunan bir eğitim verisi değeridir; bu nedenle, destek vektörlerinin sayısı hata ε değerlerindeki artışla azalır.

İkili formülasyonlarda, DVR'nin optimizasyon problemi aşağıdaki gibi temsil edilir (Smola ve Schölkopf, 2004).

$$max\ -\frac{1}{2}\sum_{i,j=1}^{n}(\alpha_i-\hat{\alpha}_i)\left(a_j-\hat{\alpha}_j\right)k\left(x_i,x_j\right)+\sum_{i=1}^{n}(\alpha_i-\hat{\alpha}_i)y_i-\varepsilon\sum_{i=1}^{n}(\alpha_i-\hat{\alpha}_1) \quad (2.11)$$

$$\sum_{i=1}^{n}(\alpha_i-\hat{\alpha}_1)=0 \quad (2.12)$$

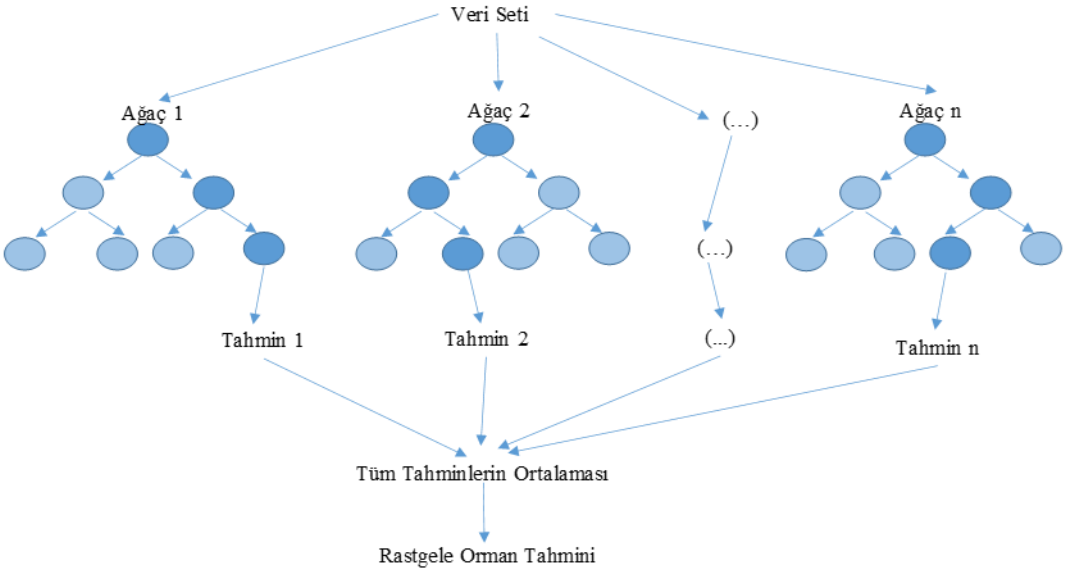
$$0\le a_i\le C, 0\le \hat{a}_1\le C \quad (2.13)$$

Burada $k\left(x_i,x_j\right)=\varphi(x_i).\varphi\left(x_j\right)$ olarak tanımlanmış olan çekirdek fonksiyonunu gösterir. φ, veri uzayından *F* özellik uzayına bir eşlemedir. α_i ve $\hat{\alpha}_1$ Lagrange çarpımlarıdır. Lagrange çarpanını ve optimallik koşullarını kullanarak regresyon fonksiyonu açıkça aşağıdaki gibi formüle edilebilir (Smola ve Schölkopf, 2004).

$$\sum_{i=1}^{n}(\alpha_i-\hat{\alpha}_1)k(x_i,x)+b \quad (2.14)$$

2.3.2. Rastgele orman

RO algoritması, sınıflandırma ve regresyon problemi için kullanılabilecek topluluk öğrenme modelidir. RO, birden fazla karar ağacı oluşturur ve daha doğru ve istikrarlı bir tahmin elde etmek için bunları birleştirir. Tahmin için, her test verisi ormandaki her karar ağacından geçirilir. Ağaçlar daha sonra bir sonuca oy verir ve tahmin, modeller arasındaki çoğunluk oylamasından üretilir ve bundan sonra daha güçlü ve daha sağlam bir tek öğrenici ile sonuçlanır (Liang ve diğerleri, 2020). Bu yapı Şekil 2.5'te gösterilmiştir (Uyanık ve diğerleri, 2020).



Şekil 2.5. RO regresyonu (Uyanık ve diğerleri, 2020)

Bir RO'da, özellikler her karar bölümünde rastgele seçilir. Ağaçlar arasındaki korelasyon, tahmin gücünü artıran ve daha yüksek verim sağlayan özelliklerin rastgele seçilmesiyle azaltılır. Bu nedenle RO algoritmasının avantajları şunlardır:

- Aşırı öğrenme sorununun üstesinden gelir.
- Eğitim verilerinde, aykırı verilere daha az duyarlıdırlar.
- Parametreler kolayca ayarlanabilir ve bu nedenle ağaçları budama ihtiyacını ortadan kaldırır.

Algoritma şu şekilde çalışır: Ormandaki her ağaç için orijinal verilerden bir önyükleme örneği seçilir. Önyükleme yapılmış örnek, orijinal veriden rastgele seçilerek yerine konulan örneklerden elde edilir ve orijinal veri seti ile aynı boyuttadır. Daha sonra bir karar ağacı, değiştirilmiş bir karar ağacı öğrenme algoritması kullanılarak önyüklenen örnek üzerinde budama yapılmadan mümkün olan maksimum ölçüde büyütülür. Ağaç-öğrenme algoritması şu şekilde değiştirilir: Her düğümde, tam özellik kümesi yerine rastgele bir özellik alt kümesi incelenerek en iyi bölme seçilir. En iyi bölmeye karar vermek öğrenme sürecinin hesaplama açısından en pahalı yönü olduğundan bir özellik alt kümesinin seçilmesi ağacın öğrenmesini büyük ölçüde hızlandıracaktır. Tüm ağaçlar bu

şekilde oluşturulduğunda, ağaçların bireysel tahminlerinin ortalaması alınarak nihai tahminler elde edilir (Nagalla ve diğerleri, 2017).

2.3.3. k-en yakın komşu

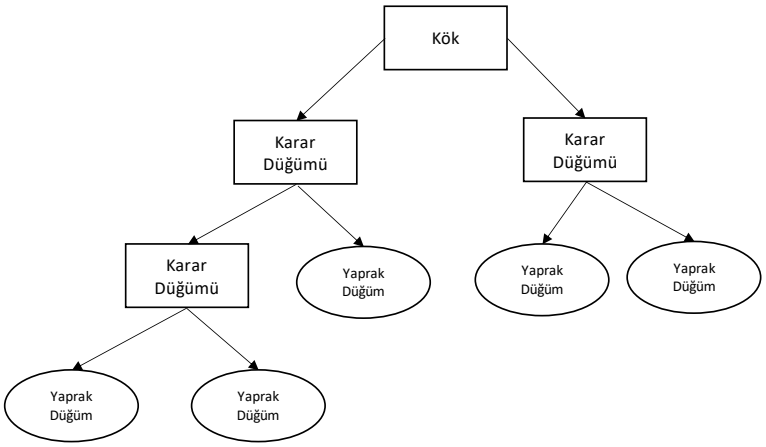
k-EK algoritması, literatürde yaygın olarak kullanılan tembel öğrenmeye dayalı sınıflandırma ve regresyon görevleri gerçekleştiren bir algoritmadır. k-EK algoritması, eğitim aşamasında belirlenen k sınıf merkezlerini dikkate alarak test değerlerinin bu sınıf merkezlerine olan uzaklığına göre sınıflandırma işlemini gerçekleştirir. Sınıf merkezlerine yakınlık ölçütü olarak Öklid, Minkowski ve Manhattan uzaklıkları gibi farklı uzaklık ölçütleri kullanılmaktadır. k-EK algoritması, rastgele k sınıf merkezi tanımlayarak algoritmayı başlatır ve eğitim verilerini bu sınıf merkezlerine yakınlıklarına göre sınıflandırır. Daha sonra, sınıf merkezlerini yinelemeli olarak eğitim verilerinin ortasına kaydırır ve yeniden sınıflandırmayı gerçekleştirir. Tatmin edici performans elde edildiğinde, k-EK algoritması sınıflandırma modelini üretir (Bozdağ ve diğerleri, 2020).
k-EK algoritması beş adımlı bir süreçtir. Bu süreçler: (a) uzaklık metriği seçilir (b) en yakın komşu sayısı seçilir (c) diğer veri noktalarından istenen noktaya olan mesafe hesaplanır (d) noktalar artan mesafe sırasına göre sıralanır (e) k en yakın komşunun yanıtlarının ortalaması hesaplanır (Kumar ve Sahu, 2021).

2.3.4. Karar ağaçları

KA, regresyon veya sınıflandırma problemlerinde kullanılır. Hem sınıflandırma hem de regresyon için bir KA kullanılıyorsa, bunlar Sınıflandırma ve Regresyon Ağacı (CART-Classification and Regression Trees) olarak adlandırılır.
KA; kök düğüm, iç düğümler, dallar ve yapraklardan oluşur. Bir düğüm, veri kümesindeki bir özniteliği (özelliği) belirtir. Kök düğüm tüm örnek kümesini temsil eder. Karar düğümü, düğümün alt düğümlere bölündüğü yerdir; bir düğüm daha fazla bölünmediğinde, o zaman bir yaprak düğüm oluşur. Bir dal, iki düğümü veya bir düğümü ve bir yaprağı birbirine bağlar. Her düğümün, ana düğümdeki olası öznitelik değeri olarak etiketlenmiş bir dizi dalı vardır. Yapraklar, sınıflandırmanın karar değeri olarak etiketlenir. Kök düğümden başlayarak, veriler özyinelemeli olarak alt kümelere bölünür. Her adımda bir kritere göre en iyi bölüm belirlenir. Yaygın olarak kullanılan kriterler gini

indeksi ve entropidir. Bu işlem, geçerli alt kümedeki tüm veriler aynı sınıfa ait olduğunda sona erer (Lin ve diğerleri, 2012).

Oluşturulan ağaç formu bir kök düğümle başlar ve kararlar karar düğümlerinde gerçekleştirilir. Şekil 2.6'da karar ağacının yapısı gösterilmektedir (Krishnan, 2021).



Şekil 2.6. KA yapısı (Krishnan, 2021)

KA uygulamak için temel sorun, her seviyedeki kök düğüm için öznitelik seçmektir. Bu problemin üstesinden gelmek için (nitelik seçimi), bilgi kazancı ve gini indeksi olmak üzere iki nitelik seçim ölçütü vardır.

KA yaklaşımının avantajları aşağıdaki gibidir (Shobha ve Rangaswamy, 2018):

- Karar ağaçlarının anlaşılması, yorumlanması ve görselleştirilmesi kolaydır.
- Değişken özellik seçimini dolaylı olarak gerçekleştirir.
- Sayısal ve kategorik veriler verimli bir şekilde işlenebilir.
- Karar ağaçları, yeni senaryolara kolayca adapte olabilmelerinden dolayı çok esnektir.
- Ağacın performansı, parametrelerin doğrusal olmayan ilişkisinden olumsuz etkilenmez.
- Veri hazırlığı, kullanıcılardan herhangi bir çaba gerektirmeden yapılabilir.

2.3.5. Ridge ve Lasso regresyon

En Küçük Mutlak Büzülme ve Seçim Operatörü (LASSO -Least Absolute Shrinkage and Selection Operator) regresyon algoritması, lineer regresyonun model karmaşıklığını azaltmak ve modelin veriye bağlı aşırı uyumunu önlemek için önerilen istatistiksel bir regresyon algoritmasıdır (Tibshirani, 2011). LASSO regresyon algoritması, girdi parametrelerinin önemini artırıp azaltarak regresyon modelinin daha iyi sonuçlar üretmesini sağlar. Bu sayede hem modele gereğinden fazla uyum sağlar hem de parametre seçimini kendi içinde yapar. LASSO regresyon algoritması, L1 düzenlileştirme yaklaşımını kullanarak katsayılarının mutlak değerinin bir oranını optimizasyon sürecine dahil eder. Bu şekilde parametrelerin sonuca etkisi düzenlenir (Bozdağ ve diğerleri, 2020).

Ridge regresyon modeli (RR), LASSO'nun L1 cezasını L2 ile değiştirir (Ribeiro, 2021). Ridge kriteri, ekstra bir cezalandırma terimi ekleyerek en küçük kareler kriteri üzerine kuruludur. Cezalandırma terimi, L2 parametre vektörünün büyüklüğü ile orantılıdır. Orantılılık katsayısı λ aynı zamanda cezalandırma parametresi olarak da adlandırılır. Ceza, en küçük kareler tahmin edicisinin katsayılarını küçültme eğilimindedir, ancak asla onları iptal etmez (Frouin ve diğerleri, 2020).

RR ve LASSO analiz yöntemleri, katsayıların tahmininde kullanılan büzülme yöntemleridir ve bazı durumlarda katsayıların tahminini en küçük kareler yönteminden daha doğru bir şekilde gerçekleştirir. RR ve LASSO' da, regresyon modelindeki tahmin edicilerin sayısının azaltılmasına izin veren “alt küme seçimi” tekniklerine bir alternatif oluşturmaktadır. Bu iki regresyon tekniğinde katsayı tahminlerinin sıfıra eşit veya sıfıra yakın olduğu doğrusal modeller üretilir (Melkumova ve Shatskikh, 2017). RR, çoklu doğrusal regresyona benzer bir yol izler, ancak en küçük kareler yöntemiyle türetilen katsayılar kullanılmaz. Her katsayının karesi bir ceza uygulanarak azaltılır. Ridge ve LASSO regresyonları arasındaki temel fark ceza şeklindedir. Düzenlileştirme parametresi, parametrelerin kareleri yerine mutlak değer olarak uygulanır (Hastie ve diğerleri, 2009).

2.3.6. Elastik net

Elastik net, sınıflandırma ve tahmin için yararlı olan, makine öğrenimine dayalı bir regresyon analizi tekniğidir. Verilerden verimli bir model geliştirmek için buraya L1 ve L2 düzenlileştirme olarak adlandırılan ve sırasıyla Lasso ve Ridge yöntemlerinde yaygın olarak kullanılan bir ceza eklenir. Lasso ve Ridge yöntemlerinden türetilen mutlak ve kare düzenlileştirmelerin yapıldığı istatistiksel bir yöntemdir. Lasso yönteminde, çeşitli parametrelerin katsayılarından elde edilen mutlak değeri temsil eden bir modele düzenlileştirme terimleri eklenir. Öte yandan, Ridge yönteminde, katsayı parametrelerinin kare fonksiyonunun eklenmesiyle düzenlileştirme yapılır. Elastik net algoritması, $0 \leq \alpha \leq 1$ değerine sahip bir "α" parametresi aracılığıyla Lasso ve Ridge algoritmaları arasında bir sönümleme gerçekleştirir. Elastik Net, sırasıyla α =1 ve α= 0 değeri için Lasso ve Ridge algoritmasına eşdeğer hale gelir. Elastik Net, aşağıdaki denklemle matematiksel olarak temsil edilir (Vakharia ve diğerleri, 2021):

$$\frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{2n} + \lambda\left(\frac{1-\alpha}{2}\sum_{j=1}^{m}\hat{\beta}_j{}^2 + \alpha\sum_{j=1}^{m}\left|\hat{\beta}_j\right|\right) \qquad (2.15)$$

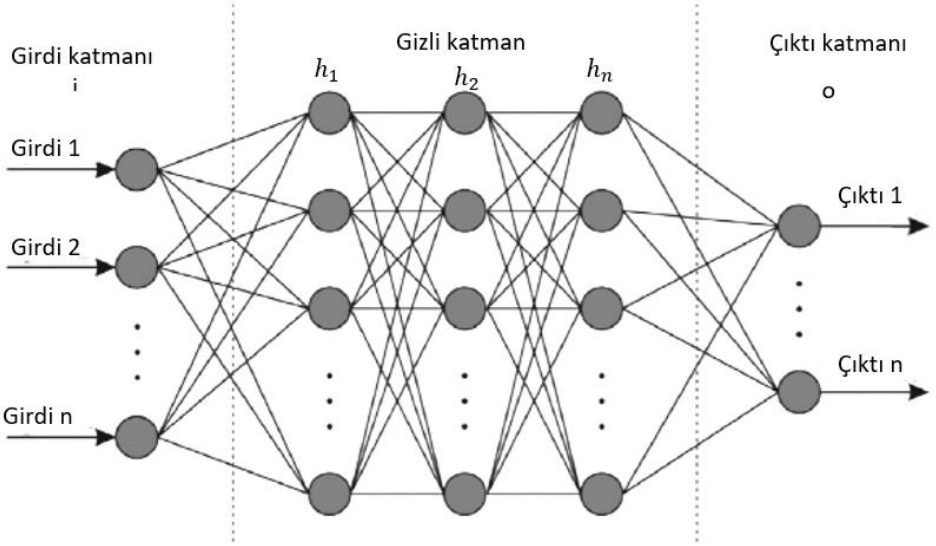
burada y_i, gerçek değeri; $\hat{y}_i$, tahmin edilen değeri ve $\hat{\beta}_j$, eğim değerini temsil eder.

2.3.7. Yapay sinir ağları

YSA, insan beynindeki nöronlardan ve bu nöronların kurduğu bağlantılardan esinlenerek geliştirilen hesaplama sistemleridir (Jain ve diğerleri, 1996).

Temel olarak ağ, bir veya daha fazla gizli katman tarafından bağlanan giriş katmanı ve çıkış katmanından oluşur. Bir YSA, nöronlar olarak da adlandırılan düğümlerden, bu nöronlar arasındaki ağın öğrenme sürecinde uyarlanabilen ağırlıklı bağlantılardan ve her bir düğümün girdi değerlerine bağlı olarak çıkış değerini tanımlayan bir aktivasyon fonksiyonundan oluşur. Aktivasyon fonksiyonu, nöronun girdi değerini dönüştürür. Gizli katmanın nöronlarında yaygın olarak kullanılan aktivasyon fonksiyonları sigmoid, hiperbolik tanjant veya diğer doğrusal olmayan fonksiyonun kullanımını içerir (Achieng, 2019). Her sinir ağı farklı katmanlardan oluşur. Giriş katmanı, ilgili veri girişinin öznitelik değerleri gibi harici kaynaklardan bilgi alır, çıkış katmanı ağın çıkışını üretir ve

gizli katmanlar, giriş ve çıkış katmanını birbirine bağlar. Her katmandaki her bir düğümün girdi değeri, düğümler arasındaki ara bağlantının ilgili ağırlığı ile çarpılan tüm girdi düğümlerin toplamı ile hesaplanır. YSA yönteminin görsel diyagramı Şekil 2.7'de gösterilmektedir (Mahmood ve diğerleri, 2021).



Şekil 2.7. YSA yapısı (Mahmood ve diğerleri, 2021)

Genellikle, giriş sinyallerinden bir nöronun ağırlıklı kombinasyon çıktısı (x_1, x_2,…, x_m) aşağıda gösterildiği gibi matematiksel olarak temsil edilir:

$$a_i' = \sum_{j=1}^{n} w_{ij}\, x_i^{l-1} + b_i^l \tag{2.16}$$

Burada, a_i' , l.gizli katmanda i. nöronun ağırlıklı kombinasyon çıktısıdır. w_{ij}, bir önceki (l -1) katmanın j. nöronu ve l. gizli katmandaki i. nöronu arasındaki ara bağlantı ağırlığıdır. x, girdiyi ve x_i^{l-1} ise l -1 katmanı veya bir önceki katmanın çıktısını gösterir. b_i^l, l. gizli katmandaki i. nöronunun eğilimidir (Achieng, 2019).

2.3.8. Çok katmanlı algılayıcı

ÇKA ağı, ileri beslemeli yapay sinir ağlarının tipik bir temsilcisidir. Girdi katmanı, gizli katman ve çıkış katmanı olmak üzere üç bölümden oluşur. Gizli ve çıkış katmanlarındaki tüm düğümler, doğrusal olmayan bir aktivasyon fonksiyonu kullanan nöronlardır. Giriş katmanı, işlenecek giriş sinyalini alır. Tahmin ve sınıflandırma gibi gerekli görevler çıktı

katmanı tarafından gerçekleştirilir. Giriş ve çıkış katmanı arasına yerleştirilen rastgele sayıda gizli katman, ÇKA'nın gerçek hesaplama motorudur. ÇKA'lar, herhangi bir sürekli fonksiyona yaklaşmak için tasarlanmıştır ve doğrusal olarak ayrılamayan sorunları çözebilir. ÇKA'nın başlıca kullanım durumları, örüntü sınıflandırması, tanıma, tahmindir (Abirami ve Chitra, 2020).

2.4. Literatür Taraması

Hava kirliliği sorunlarını yönetmek ve çözmek uzun vadeli bir süreçtir. Hava kalitesi tahmini, hava kirliliğinin neden olduğu hasarın önlenmesine yardımcı olabilir. Bu nedenle, koruyucu önlemler alınması ve ciddi kirlilik olayların önlemesi için hava kalitesi tahminlerinin zamanında yapılması gereklidir.

Makine öğrenimi, son zamanlarda büyük verilerle güçlü ve hızlı tahminleri için muazzam bir popülerlik kazanmıştır. Bazı araştırmacılar, hava kalitesinin kısa ve uzun vadeli tahmini için başarılı bir şekilde makine öğrenmesi algoritmalarını uygulamışlardır (Liu ve diğerleri, 2019).

Bozdağ vd. (2020) çalışmalarında, makine öğrenmesi algoritmalarından LASSO, DVM, RO, k-EK, ekstrem gradyan arttırma (XGBoost-eXtreme Gradient Boosting) ve YSA kullanmışlardır. Ankara'daki altı istasyonun 2009-2017 yıllarındaki PM_{10} konsantrasyonlarını girdi olarak vermişler ve yedinci istasyonun 2018 yılı için PM_{10} konsantrasyon değerini tahmin etmişlerdir. Model geliştirme aşamasını her istasyon için tekrar ederek ve algoritmaların ürettiği sonuçlar ile gerçek sonuçları karşılaştırarak algoritmaların performans ve hata oranlarını belirlemişlerdir. En iyi sonuçları YSA ile sağlamışlardır.

Doreswamy vd. (2020) çalışmalarında, Ocak 2012 ve Aralık 2017 arasında Taiwan hava kalitesi izleme veri setindeki hava kirleticileri ve meteorolojik parametreleri kullanarak havadaki partükül madde $PM_{2.5}$ konsantrasyon değerini makine öğrenmesi yöntemlerinden DR, RO, gradyan artırma (GR- Gradient Boosting), k-EK, CART ve ÇKA ile tahmin ederek sonuçları geleneksel yöntemle karşılaştırmışlardır. Makine öğrenmesi algoritmalarının tahmin performansında daha iyi sonuçlar verdiğini

görmüşlerdir. En iyi modelin seçiminde çapraz doğrulama kullanmışlardır. Bu modellerin performansını KOKH, OMH, OKH ve R^2 gibi istatiksel ölçümlerle değerlendirmişlerdir. Şahin vd. (2020), dokuz farklı çevresel faktör; azot oksitler (NO_x), kükürt dioksit (SO_2), azot monoksit (NO), azot dioksit (NO_2), ozon (O_3), hava sıcaklığı, hava nemi, hava basıncı ve PM_{10} değerindeki rüzgar hızını incelemişlerdir. Bu amaçla, dört katmanlı ileri beslemeli sinir ağı mimarilerini kullanmışlardır. PM_{10} değerini, bu çevresel faktörlerin ileri beslemeli sinir ağında girdi olarak kullanılmasıyla tahmin etmişlerdir. Ayrıca, az sayıda girdi parametresinin sonuçlarını görmek için temel bileşen analizi (PCA-principal component analysis) ile özelliklerin boyutunu küçültülerek deneyler yapmışlardır. Kamu kaynaklarından günlük olarak elde edilen beş aylık verileri, deneyleri yapmak için veri seti olarak kullanmışlardır. Performans ölçütleri olarak KOKH, OMH, OKH ve R^2 kullanmışlardır.

Stafoggia vd. (2019), İtalya'da $PM_{2.5}$ ve PM_{10} konsantrasyonlarının tahmini için RO yöntemini kullanmışlardır.

Li vd. (2020) çalışmalarında, Hong Kong'daki üç istasyondan toplanan verilerdeki saatlik $PM_{2.5}$ ve NOX konsantrasyonlarını tahmin etmek için RO, artırılmış regresyon ağaçları (BRT- Boosted Regresyon Trees), DVM, XGBoost dahil olmak üzere altı yaygın makine öğrenme algoritmasının performansını değerlendirmişler ve karşılaştırmışlardır.

Liu vd. (2019), Pekin'deki hava kalitesi indeksini ve İtalya şehrindeki NOX konsantrasyonunu halka açık iki veri setine dayanarak tahmin etmek için DVR ve RO regresyon kullanmışlardır. Regresyon modellerinin performansını değerlendirmek için KOKH, korelasyon katsayısı (r) ve R^2 kullanmışlardır. DVR tabanlı modelin hava kalitesi indeksi tahmininde daha iyi performans gösterdiği ve RO regresyon modelin NOX konsantrasyonunun tahmininde daha iyi performans gösterdiği sonucuna varmışlardır.

Díaz-Robles vd. (2008), Şili'deki PM_{10} ölçümlerini tahmin etmek için YSA ve bütünleşik otoregresif hareketli ortalama (ARIMA) içeren bir hibrit modelin uygulama çalışmasını yapmışlardır.

Choubin vd. (2020), Barselona'da PM_{10} kirleticisinin tahmini için makine öğrenmesi algoritmalarından RO, torbalı sınıflandırma , regresyon ağaç (Bagged-CART) ve karışım ayrım analizi (MDA-Mixture Discriminate Analysis) kullanmışlardır.

Ortiz-García vd. (2010), Madrid kentsel bölgesindeki saatlik ozon değerlerinin tahmininde DVR kullanmışlardır. Ayrıca ÇKA kullanılarak elde edilen sonuçlarda karşılaştırma yapılmıştır.

Sousa vd. (2007), ileri beslemeli YSA yöntemine dayalı yeni bir yöntem kullanarak bir sonraki günün saatlik ozon konsantrasyonlarını tahmin etmişlerdir. Geliştirilen modeli, çoklu doğrusal regresyon, ileri beslemeli YSA ve ayrıca temel bileşen regresyonu kullanarak karşılaştırmışlardır.

Chelani (2010), DVM kullanarak günlük maksimum ozon konsantrasyonlarının tahmini üzerine çalışma yapmıştır. Bunun için 2002 2004 yılında Delhi'de bir sahada gözlemlenen günlük maksimum ozon konsantrasyonu verilerini ve mevcut meteorolojik parametreleri kullanmıştır. Performans değerlendirmek için DVM ve YSA algoritmaları arasında karşılaştırma yapmıştır, Sonuçlar, günlük maksimum ozon konsantrasyonlarını tahmin etmede YSA üzerinden DVM'nin umut verici performansını göstermiştir.

Jumin vd. (2020), Malezya'da bulunan büyük şehirlerde ozon konsantrasyonunu tahmin etmek için makine öğrenimi algoritmalarından DR, YSA ve güçlendirilmiş karar ağacını kullanmışlardır. Önerilen modeller, 24 saatlik ve 12 saatlik troposferik ozon konsantrasyonunu tahmin etmek için girdi olarak üç yıllık geçmiş veriler kullanılarak geliştirilmiştir. Güçlendirilmiş karar ağacı, tüm istasyonlar için doğrusal regresyon ve YSA algoritmalarından daha iyi performans göstermiştir.

Aljanabi vd. (2020), Ürdün Amman'da bir önceki günün meteorolojik ve mevsimsel değişkenlerinin bir karışımını kullanarak bir sonraki günün ozon konsantrasyonunu tahmin etmek için ÇKA, DVR, KA regresyonu ve XGBoost algoritmalarını kullanmışlardır. Ayrıca zaman serisi verileri üzerinde hareketli ortalama, Holt-Winters ve Savitzky-Golay yöntemlerini uygulamışlardır. ÇKA yönteminin diğer yöntemlerden daha iyi performans gösterdiğini ve Savitzky-Golay kullanmanın sonuçları iyileştirdiği tespit edilmiştir.

Chattopadhyay (2012), tahmin ediciler olarak bulut örtüsü, ortalama sıcaklık ve yağışı kullanarak önceki aylardaki tahmin edicilerin değerlerinden aylık toplam ozon konsantrasyonlarının tahmini için sigmoid doğrusal olmayan çok katmanlı bir algılayıcı şeklinde bir yapay sinir ağı geliştirmişlerdir. Sinir ağı modelinin geliştirilmesinden önce temel bileşen analizi yoluyla çoklu doğrusallığı ortadan kaldırmışlardır. Temel bileşen analizi ile çıkarılan değişkenlere dayanarak üç yapay sinir ağı modeli geliştirmişlerdir. Genel olarak, yapay sinir ağı meteorolojik tahmin ediciler temelinde aylık toplam ozonu tahmin etmek için iyi bir potansiyele sahip olduğunu görmüşlerdir.

Chattopadhyay vd. (2019) çalışmalarında, gradyanlı iniş öğrenmeli ÇKA yöntemi kullanmışlardır. Hindistan'da 2016 ve 2017 yıllarının muson öncesi sezonunda NOX, SO_2, PM_{10} ve sıcaklığı girdi olarak kullanarak O_3 kirleticisini tahmin etmişlerdir.

Faleh vd. (2017), Tunus şehrinde ozon konsantrasyonlarının tahmini için DVM ve k-EK yöntemlerini kullanmışlardır. k-EK yönteminde sonuçlar %98.7 başarı oranına ulaşmıştır. Bir sınıflandırıcı oluşturmak için doğrusal, polinom ve RBF çekirdeğine sahip DVM uygulamışlar ve RBF çekirdeği ile tekrar tam doğruluk (%100) elde etmişlerdir.

Capilla (2016), İspanya'nın Valensiya şehri için çoklu doğrusal regresyon ve sinir ağları modellerinin uygulamasını ele almıştır. Bu modeller, kısa vadeli tahmin aralıkları (1, 8 ve 24 saat önceden) için saatlik ozon seviyelerini tahmin etmiştir. Çalışma dönemi 2010-2012'dir. Girdi değişkenleri olarak meteorolojik gözlemler, ozon ve nitrojen oksit konsantrasyonları kullanmıştır. Sonuçların doğruluğunu değerlendirmek için performans kriterleri olarak OKH, OMH ve gözlemler ile tahminler arasındaki korelasyon katsayısı kullanmıştır. Bu kriterler, tüm konumlarda 1 saat ve 24 saat tahminleri için daha iyi sonuçlara sahip olmuştur. Çoklu doğrusal regresyon ve ÇKA ağlarının karşılaştırılması, ikinci yaklaşımın üç tahmin aralığı için daha doğru tahmin elde etmeye izin verdiğini göstermiştir.

3. MATERYAL ve YÖNTEM

3.1. Veri Tanımı

Bu çalışmada kullanılan veriler, Ulusal Hava Kalitesi İzleme Ağı web sitesinin (Çevre, Şehircilik ve İklim Değişikliği Bakanlığı, 2022) ölçüm sonuçlarından elde edilmiştir. Veriler, 1 Ocak 2016 ile 28 Mayıs 2021 tarihleri arasında Bursa ili için Bursa Uludağ Üniversitesi ve Kültürpark istasyonlarının saatlik hava kirliliği ve meteorolojik verilerini içerir. Veri seti 11 öznitelikten ($PM_{2.5}$, SO_2, NO, NO_2, NOX, O_3, hava sıcaklığı, rüzgar yönü, rüzgar hızı, bağıl nem, hava basıncı) ve toplam 47393 örnekten oluşmaktadır. Detaylar EK 1'de gösterilmiştir.

3.2. Veri Ön İşleme

Veri kalitesi ve temsili, tahmin modellerinin başarılı bir şekilde oluşturulmasını garanti eden ilk ve en önemli noktalardır. Veri işleme, verileri topladıktan sonra verileri uygun bir formata dönüştürme yeteneğidir. Veri ön işleme adımı, genellikle bir makine öğrenimi algoritmasının genelleme yeteneğini etkiler (Kotsiantis ve Kanellopoulos, 2006). Veri ön işleme genellikle eksik veri atamasını, aykırı gözlemleri kaldırmayı veya değiştirmeyi, veri dönüştürmeyi (genellikle normalleştirme ve standardizasyon) ve öznitelik mühendisliğini kapsar. Veri işleme ile bu veriler algoritmalara uygun hale getirilir (Castelli ve diğerleri, 2020).

Veri ön işleme Python programlama dili kullanılarak gerçekleştirilmiştir. İlk olarak veri temizleme çalışması yapılmıştır. Veri setinde boş değer olup olmadığı kontrol edilmiştir ve eksik değer problemi ortalama değer ile doldurma yöntemiyle çözülmüştür. Veri kümesinin detayları EK 1'de sunulmuştur. Daha sonra bir kutu grafiği ve z skor yöntemleri kullanarak veri setindeki aykırı değerler tespit edilmiştir. Aykırı değerleri azaltmak için silme yaklaşımı kullanılmıştır.

3.3. Modellerin Geliştirilmesi

Makine öğrenimi modelinin ayarlanması gereken birçok parametresi vardır ve bu parametrelerini değiştirerek modelin performansı iyileştirilebilir. Hiper parametre ayarlama, bir sınıflandırıcının performansını değerlendirmek ve farklı sayıda parametre kombinasyonu yürütmek için en iyi yöntemdir. Bir sınıflandırıcının eğitim verilerini

kullanarak değerlendirmek, aşırı öğrenme olarak adlandırılan temel bir makine öğrenimi sorununa neden olacaktır. Aşırı öğrenme, bir modelin test verilerinde düşük ve eğitim verilerinde yüksek performans gösterdiği durumdur. Bu nedenle, hiper parametre optimizasyonu için grid search yöntemiyle çapraz doğrulama kullanılır.

Kullanılan makine öğrenmesi regresyon algoritmaları; RO, KA, DV, k-EK ve ÇKA regresyondur. Saatlik olarak ölçülen kirlilik parametreleri ($PM_{2.5}$, SO_2, NO, NO_2, NOX) ve beş farklı meteorolojik faktör (hava sıcaklığı, rüzgar yönü, rüzgar hızı, bağıl nem, hava basıncı) modelin girdi parametreleri olarak kullanılıp çıktı olarak O_3 konsantrasyonu tahmin edilmiştir. Modellerimizi değerlendirmek için holdout yöntemi kullanılmıştır. Bu yönteme göre, tüm veri seti iki parçaya bölünür, eğitim bölümü (%80) modeli eğitmek için kullanılır ve test bölümü (%20) modeli test etmek için kullanılır. Veri kümelerinin %80'i modelin eğitim amaçları için keyfi olarak atandıktan sonra, kalan kısım modelin yeterliliğini değerlendirmek için kullanılmıştır.

Uygulanan tahmin yöntemlerinin başarısını değerlendirmek ve aşırı öğrenmeyi önlemek için k-kat çapraz doğrulama yöntemi kullanılmıştır. Burada veriler, ilk katın doğrulama yani test seti olarak kullanıldığı, diğer k-1 katların ise eğitim seti için kullanıldığı k kat sayısına bölünür. k iterasyon sayısı beş olarak belirlenmiştir. Model geliştirmelerde kullanılan kodlar EK 2'de gösterilmiştir.

3.3.1. Hiper parametre ayarlama

Hiper parametre optimizasyonu için kullanılan parametreler aşağıdaki gibidir:

- k-en yakın komşu algoritması için n_neighbors parametresi komşu sayısını gösterir.
- Karar ağacı ve rassal orman algoritmasında max_depth, ağacın maksimum derinliğini gösterir, min_samples_split, bir iç düğümü bölmek için gereken minimum örnek sayısıdır. Çok küçük bir değer, aşırı öğrenmeye neden olurken, büyük bir değer muhtemelen eksik öğrenmeye neden olur. Min_samples_leaf, bir yaprak düğümde olması gereken minimum örnek sayısını gösterir.
- Destek vektör makineleri algoritmasında C, düzenlileştirme parametresidir. Düzenlileştirmenin gücü C ile ters orantılıdır. Kesinlikle pozitif değer almalıdır.

- Çok katmanlı algılayıcı algoritmasında hidden_layer_sizes, i. gizli katmandaki nöronların sayısını temsil eder, α ise L2 ceza parametresini gösterir.

3.4. Modellerin Performans Değerlendirmesi

Algoritmalar Jupyter Notebook ortamında Python programlama dili kullanılarak kodlanmış ve Pandas kullanılarak ön işleme ve zaman serisi değerlendirmesi yapılmıştır. Makine öğrenimi algoritmalarında, açık kaynaklı bir makine öğrenimi kitaplığı olan scikit öğrenme kitaplığı kullanılmıştır. Grafikleri çizmek için plot kitaplığı kullanılmıştır. Performansın değerlendirilmesi scikit-learn metrikleri kullanılarak yapılmıştır. Tüm hiper parametreler, beş kat çapraz doğrulama yöntemi ve Grid SearchCV işlevi kullanılarak ayarlanmıştır. EK 2'de algoritmaların Python kodları verilmiştir

3.4.1. Hata metrikleri

Tahmin modelinin veri üzerindeki performansını değerlendirmek ve model tarafından tahmin edilen değer ile gerçek değer arasındaki ilişkiyi ölçmek için farklı hata metrikleri kullanılır. Bu çalışmada, tahmin modellerinin değerlendirilmesi için KOKH, OKH, OMH, OMYH ve R^2 kullanılmıştır. Bu hata ölçütleri içerisinde, tahmin modelinin doğruluk oranı kriteri R^2 yöntemi ile yapılırken, algoritmanın hata ölçüsü OMH, KOKH, OKH, OMYH ile değerlendirilmektedir. Bu metrikler 0 ile ∞ arasında değerler alabilir. Ancak, metriklerin hesaplamalarından elde edilen değerler küçüldükçe tahmin algoritmasının daha başarılı olduğu belirlenmiştir. KOKH, kare işlevinden önceki ortalama nedeniyle büyük veri kümesinde daha avantajlıdır. Ayrıca R^2 ile ölçülen doğruluk oranı 1'e yaklaştıkça tahmin modelinin başarısının daha yüksek olduğu anlaşılmaktadır (Wang ve Xu, 2004).

3.4.1.1. Ortalama mutlak hata

OMH, gerçek değer ile tahmin edilen değer arasındaki mutlak hataya karşılık gelir ve herhangi bir yön dikkate alınmadan verideki (tahminler) hataların ortalama büyüklüğünü ölçmek için kullanılır.

$$\text{OMH} = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i| \tag{3.1}$$

3.4.1.2. Kök ortalama kare hata

Regresyon eğrisinde tahmin edilen değerler ile gerçek değerleri arasındaki mesafenin ölçüsü artıkları ifade eder ve artıkların standart sapması, hatanın ortalama karekökünü verir.

$$\text{KOKH}= \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \tag{3.2}$$

3.4.1.3. Açıklayıcılık katsayısı

Açıklayıcılık katsayısı R^2, tahminin doğruluğunu temsil eder. 1'e yakın R^2 değeri daha fazla tahmin doğruluğu anlamına gelir.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \quad \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i \tag{3.3}$$

3.4.1.4. Ortalama mutlak yüzde hata

OMYH, hatanın büyüklüğünü yüzde olarak ölçer.

$$\text{OMYH} = \frac{1}{n}\sum_{i=1}^{n}\frac{|\hat{y}_i - y_i|}{y_i} \tag{3.4}$$

3.4.1.5. Ortalama kare hata

OKH, Gerçek değer ile tahmin edilen değer arasındaki ortalama kare farkını ifade eder.

$$\text{OKH} = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2 \tag{3.5}$$

4. BULGULAR ve TARTIŞMA

Bir veya daha fazla değere ve aykırı değerlere sahip veriler kaldırıldıktan sonra, O_3 tahmini için Bursa Uludağ Üniversitesi ve Kültürpark istasyonları için sırasıyla 28514 ve 11408 veri örneği elde edilmiştir. İki istasyon için kullanılan veri kümeleri, modeli eğitmek ve model performansını değerlendirmek için rastgele eğitim ve test örneklerine bölünmüştür. Veri örneklerinin %80'i kullanılarak eğitilmiştir ve veri örneklerinin geri kalan %20'si test için ayrılmıştır. Her bir regresyon yönteminin performansı, çapraz doğrulama yaklaşımıyla grid search yoluyla bir hiper parametre alanı seçilerek test setinin aşırı öğrenme riskini en aza indirecek şekilde optimize edilmiştir. Izgara arama (Grid Search) için eğitim veri setindeki verilerin %80'i üzerinde beş katlı çapraz doğrulama gerçekleştirilmiştir. Eğitilen modeller her bir hata metrik yöntemi ile değerlendirilmiştir.

4.1. Bursa Uludağ Üniversitesi İstasyon Sonuçları

Kullanılan makine öğrenmesi regresyon algoritmaları için test setleri kullanarak bulunan hata metrikleri Uludağ Üniversitesi istasyonu için Çizelge 4.1'de özetlenmiştir.

Çizelge 4.1. Bursa Uludağ Üniversitesi istasyonu için O_3 tahmininde kullanılan makine öğrenmesi algoritmalarının sonuçları

Algoritmalar	KOKH	OKH	R^2	OMH	OMYH
Karar Ağacı	15,452	238,765	0,834	11,684	0,506
Rastgele Orman	13,741	188,811	0,869	10,405	0,468
Destek Vektör Makinesi	17,534	307,454	0,786	13,430	0,585
k-en yakın komşu	15,757	248,281	0,827	11,783	0,53
Çok katmanlı algılayıcı	13,827	191,173	0,867	10,665	0,502

4.1.1. Karar ağacı regresyon sonuçları

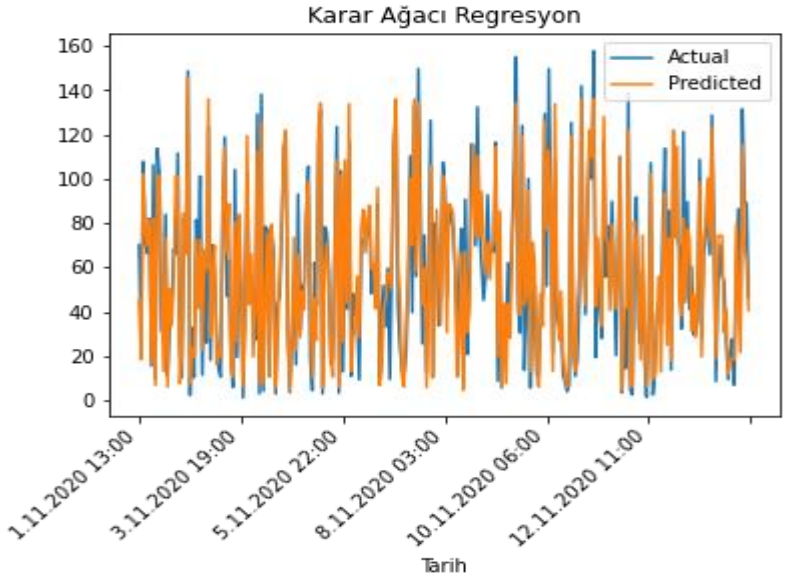
KA regresyon modellemesinin optimum değerleri Python'daki ızgara arama (Grid Search) işlevi kullanılarak beş kat çapraz doğrulama tekniği ile elde edildi. KA algoritması için kullanılan parametre değerleri ve optimum sonuçlar Çizelge 4.2'de gösterilmiştir. Model için en iyi kombinasyonun max_dept parametresinin 10, min_samples_leaf parametresinin 20 ve min_samples_split parametresinin 2 olduğu durumda bulunmuştur. Test seti için bulunan hata metrikleri Çizelge 4.1'de özetlenmiştir.

KOKH değeri 15,452, OKH değeri 238,765, OMH değeri 11,684, OMYH değeri 0,506 ve R^2 değeri 0,834 bulunmuştur.

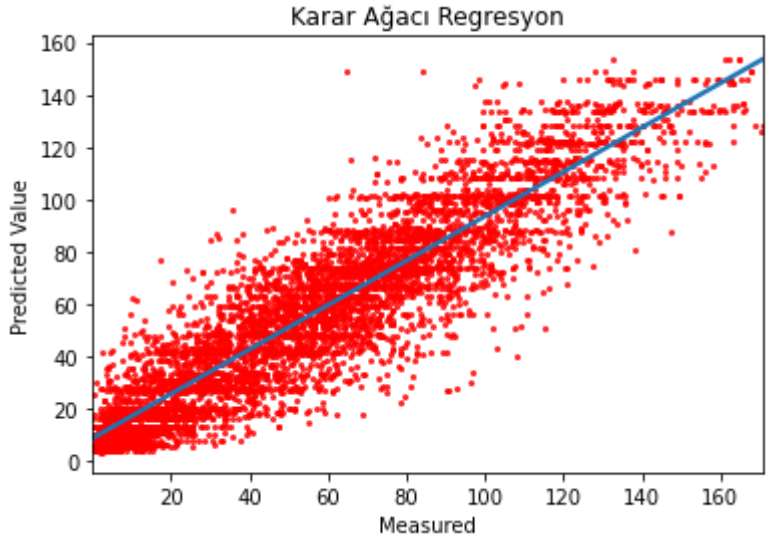
Çizelge 4.2. Bursa Uludağ Üniversitesi istasyonu için KA regresyonda GridSearchCV kullanarak en iyi parametreleri ayarlama

Parametreler	Değerler	En iyi değer
max_depth	[5, 8, 10]	10
min_samples_leaf	[2, 10, 80, 100]	20
min_samples_split	[10, 15, 20]	2

KA regresyon modellemesi kullanarak gözlenen değerlere karşı tahmin edilen O_3 konsantrasyonunun grafikleri Şekil 4.1 ve Şekil 4.2'de gösterilmektedir.



Şekil 4.1. Bursa Uludağ Üniversitesi istasyonu için 1-12 Kasım tarihleri arasında KA regresyon kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerin karşılaştırma grafiği



Şekil 4.2. Bursa Uludağ Üniversitesi istasyonu için KA regresyon kullanarak tahmin edilen saatlik O_3 değerlerin dağılım grafiği

4.1.2. Rastgele orman regresyon sonuçları

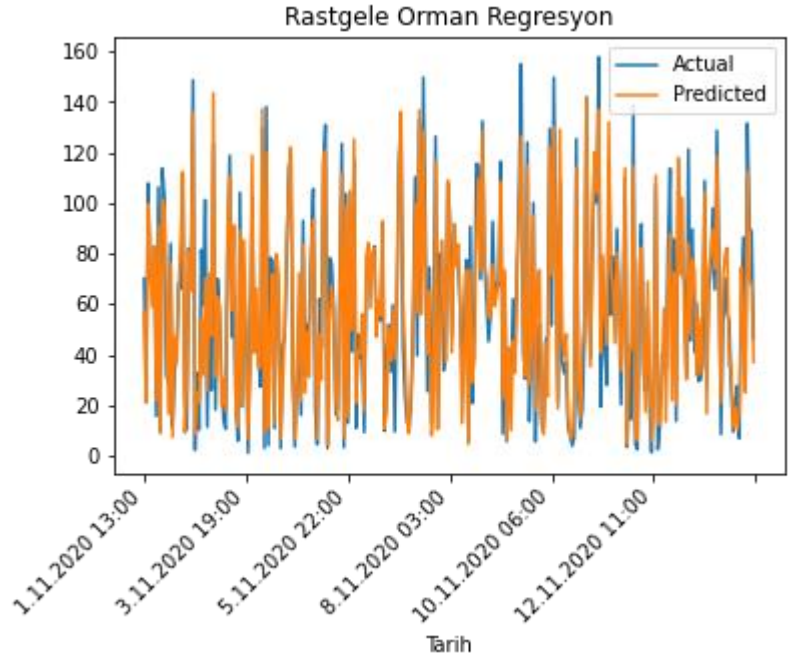
RO modellemesinin optimum değerleri Python'daki ızgara arama (Grid Search) işlevi kullanılarak beş kat çapraz doğrulama tekniği ile elde edildi. RO regresyon algoritması için kullanılan parametre değerleri ve optimum sonuçlar Çizelge 4.3'te gösterilmiştir. Model için en iyi kombinasyonun max_dept parametresinin 10, min_samples_leaf parametresinin 10 ve min_samples_split parametresinin 10 olduğu durumda bulunmuştur.

Test seti için bulunan hata metrikleri Çizelge 4.1'de özetlenmiştir. KOKH değeri 13,741, OKH değeri 188,811, OMH değeri 10,405, OMYH değeri 0,468 ve R^2 değeri 0,869 bulunmuştur.

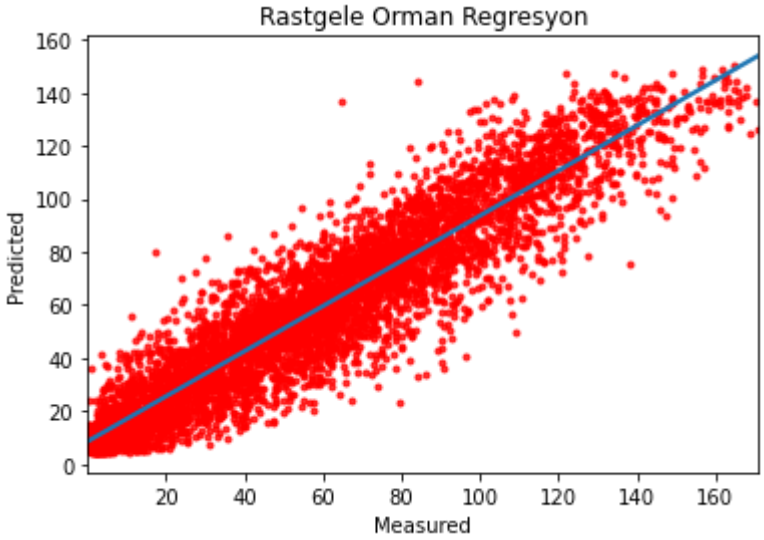
Çizelge 4.3. Bursa Uludağ Üniversitesi istasyonu için RO regresyonda GridSearchCV kullanarak en iyi parametre değerlerini ayarlama

Parametreler	Değer	En iyi değer
max_depth	[5, 8, 10]	10
min_samples_leaf	[2, 10, 80, 100]	10
min_samples_split	[10,15,20]	10

RO regresyon modellemesi kullanarak gözlenen değerlere karşı tahmin edilen O_3 konsantrasyonunun grafikleri Şekil 4.3 ve Şekil 4.4'te gösterilmektedir.



Şekil 4.3. Bursa Uludağ Üniversitesi istasyonu için 1-12 Kasım tarihleri arasında RO regresyon kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerin karşılaştırma grafiği



Şekil 4.4. Bursa Uludağ Üniversitesi istasyonu için RO regresyonu kullanarak tahmin edilen saatlik O_3 değerlerin dağılım grafiği

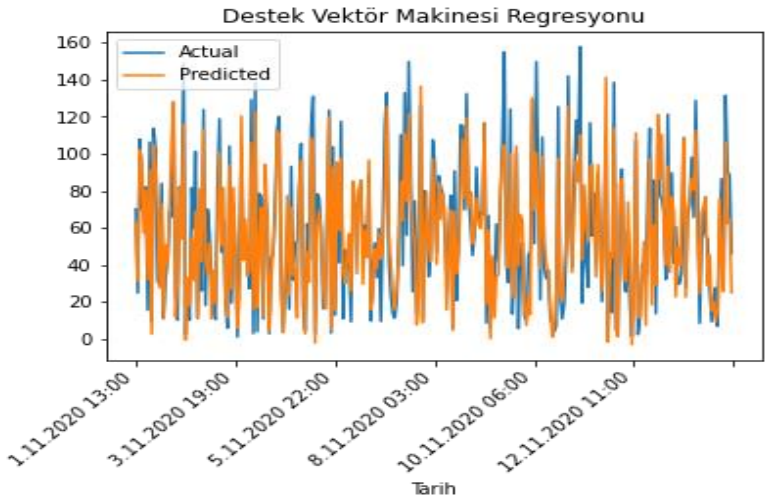
4.1.3. Destek vektör makinesi regresyon sonuçları

DVR modellemesinin optimum değerleri Python'daki ızgara arama (Grid Search) işlevi kullanılarak beş kat çapraz doğrulama tekniği ile elde edildi. DVR algoritması için kullanılan parametre değerleri ve optimum sonuçlar Çizelge 4.4'te gösterilmiştir. Deneysel sonuçlar, DVR modeli için en iyi kombinasyonun C parametresinin 1000 olduğunu göstermiştir. Test seti için bulunan hata metrikleri Çizelge 4.1'de özetlenmiştir. KOKH değeri 17,534, OKH değeri 307,454, OMH değeri 13,430, OMYH değeri 0,585 ve R^2 değeri 0,786 bulunmuştur.

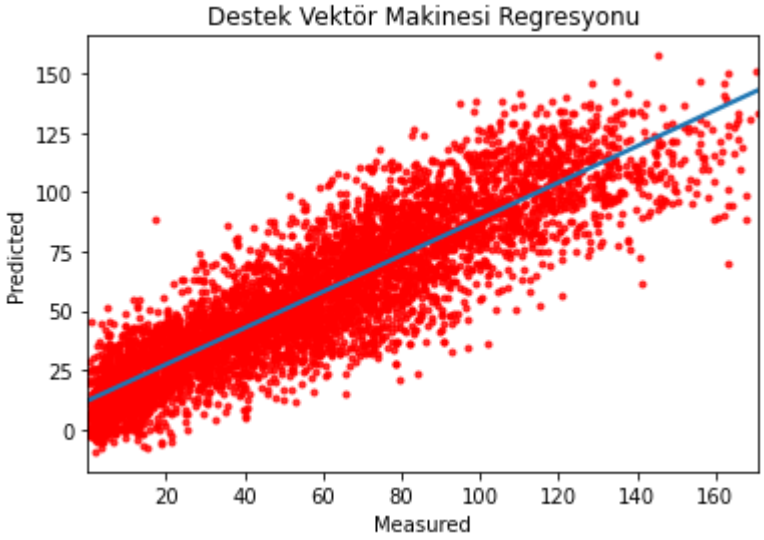
Çizelge 4.4. Bursa Uludağ Üniversitesi istasyonu için DVR algoritmasında GridSearchCV kullanarak en iyi parametre değerlerini ayarlama

Parametreler	Değer	En iyi değer
C	[0.1,1,10,100,1000]	1000

DVR modellemesi kullanarak gözlenen değerlere karşı tahmin edilen O_3 konsantrasyonunun grafikleri Şekil 4.5 ve Şekil 4.6'da gösterilmektedir.



Şekil 4.5. Bursa Uludağ Üniversitesi istasyonu için 1-12 Kasım tarihleri arasında DVR kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerin karşılaştırma grafiği



Şekil 4.6. Bursa Uludağ Üniversitesi istasyonu için DVR kullanarak tahmin edilen saatlik O_3 değerlerin dağılım grafiği

4.1.4. Çok katmanlı algılayıcı regresyon sonuçları

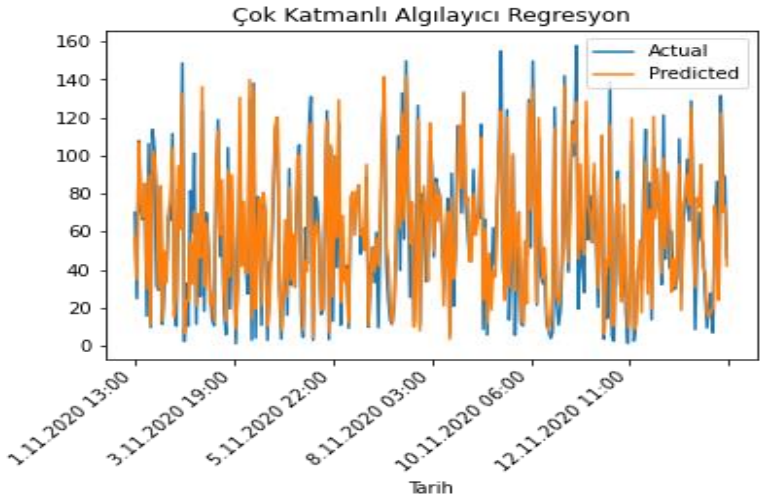
ÇKA modellemesinin optimum değerleri Python'daki ızgara arama (Grid Search) işlevi kullanılarak beş kat çapraz doğrulama tekniği ile elde edildi. Algoritma için kullanılan parametre değerleri ve optimum sonuçlar Çizelge 4.5 'te gösterilmiştir. Model için en iyi kombinasyonun alpha parametresinin 0,0001, hidden_layer_sizes parametresinin (100, 100) olduğu durumda bulunmuştur. Test seti için bulunan hata metrikleri Çizelge 4.1'de

özetlenmiştir. KOKH değeri 13,827, OKH değeri 191,173, OMH değeri 10,665, OMYH değeri 0,502 ve R^2 değeri 0,867 bulunmuştur.

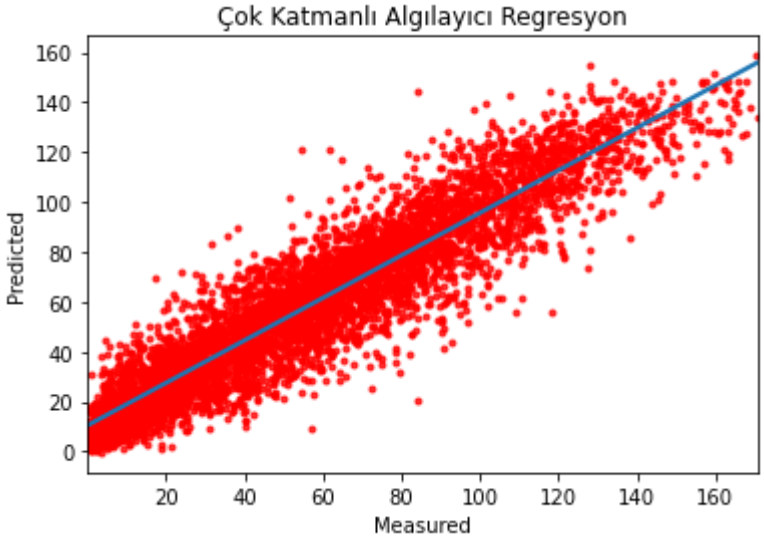
Çizelge 4.5. Bursa Uludağ Üniversitesi istasyonu için ÇKA regresyonda GridSearchCV kullanarak en iyi parametre değerlerini ayarlama

Parametreler	Değerler	En iyi değer
alpha	[0,1, 0,01, 0,02, 0,001, 0,0001]	0,0001
hidden_layer_sizes	[(10,20), (5,5), (100,100)]	(100, 100)

ÇKA regresyon modellemesi kullanarak gözlenen değerlere karşı tahmin edilen O_3 konsantrasyonunun grafikleri Şekil 4.7 ve Şekil 4.8'de gösterilmektedir.



Şekil 4.7. Bursa Uludağ Üniversitesi istasyonu için 1-12 Kasım tarihleri arasında ÇKA regresyon kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerin karşılaştırma grafiği



Şekil 4.8. Bursa Uludağ Üniversitesi istasyonu için ÇKA regresyon kullanarak tahmin edilen saatlik O_3 değerlerin dağılım grafiği

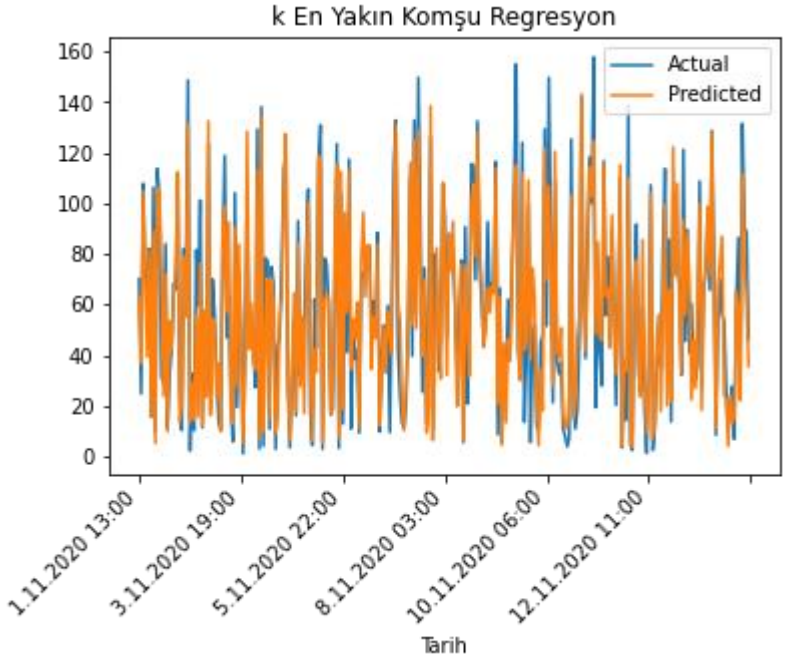
4.1.5. k-en yakın komşu regresyon sonuçları

k-EK modellemesinin optimum değerleri Python'daki ızgara arama (Grid Search) işlevi kullanılarak beş kat çapraz doğrulama tekniği ile elde edildi. Algoritma için kullanılan parametre değeri ve optimum sonuç Çizelge 4.6'da gösterilmiştir. Model için en iyi sonuç n_neighbors parametresinin 10 olduğu durumda bulunmuştur. Test seti için bulunan hata metrikleri Çizelge 4.1'de özetlenmiştir. KOKH değeri 15,757, OKH değeri 248,281, OMH değeri 11,783, OMYH değeri 0,530 ve R^2 değeri 0,827 bulunmuştur.

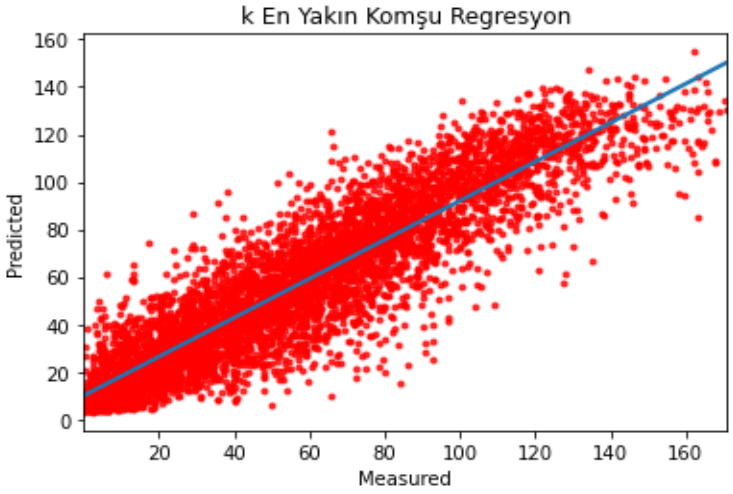
Çizelge 4.6. Bursa Uludağ Üniversitesi için k-EK regresyonda GridSearchCV kullanarak en iyi parametre değerlerini ayarlama

Parametreler	Değerler	En iyi değer
n_neighbors	(1, 30, 1)	10

k-EK regresyon modellemesi kullanarak gözlenen değerlere karşı tahmin edilen O_3 konsantrasyonunun grafikleri Şekil 4.9 ve Şekil 4.10'da gösterilmektedir.



Şekil 4.9. Bursa Uludağ Üniversitesi istasyonu için 1-12 Kasım tarihleri arasında k-EK regresyon kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerin karşılaştırma grafiği



Şekil 4.10. Bursa Uludağ Üniversitesi istasyonu için k-EK regresyon kullanarak tahmin edilen saatlik O_3 değerlerin dağılım grafiği

4.2. Kültürpark İstasyon Sonuçları

Kullanılan makine öğrenmesi regresyon algoritmaları için test setleri kullanarak bulunan hata metrikleri Kültürpark istasyonu için Çizelge 4.7'de özetlenmiştir.

Çizelge 4.7. Kültürpark istasyonu için O_3 tahmininde kullanılan makine öğrenmesi algoritmalarının sonuçları

Algoritmalar	KOKH	OKH	R^2	OMH	OMYH
Karar Ağacı	13,922	193,835	0,849	9,786	0,581
Rastgele Orman	11,816	139,617	0,891	8,443	0,559
Destek Vektör Makinesi	17,878	319,608	0,751	12,886	0,957
k-en yakın komşu	13,763	189,422	0,852	9,512	0,664
Çok katmanlı algılayıcı	13,201	174,257	0,864	9,606	0,654

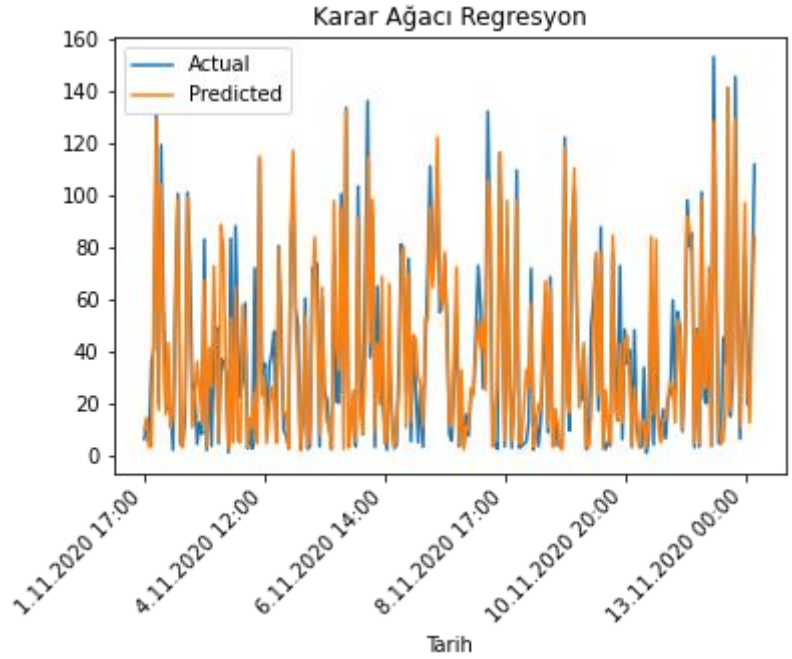
4.2.1. Karar ağacı regresyon sonuçları

KA regresyon modellemesinin optimum değerleri Python'daki ızgara arama (Grid Search) işlevi kullanılarak beş kat çapraz doğrulama tekniği ile elde edildi. KA algoritması için kullanılan parametre değerleri ve optimum sonuçlar Çizelge 4.8'de gösterilmiştir. Model için en iyi kombinasyonun max_depth parametresinin 10, min_samples_leaf parametresinin 15, min_samples_split parametresinin 2 olduğu durumda bulunmuştur. Test seti için bulunan hata metrikleri Çizelge 4.7'de özetlenmiştir. KOKH değeri 13,922, OKH değeri 193,835, OMH değeri 9,786, OMYH değeri 0,581 ve R^2 değeri 0,849 bulunmuştur.

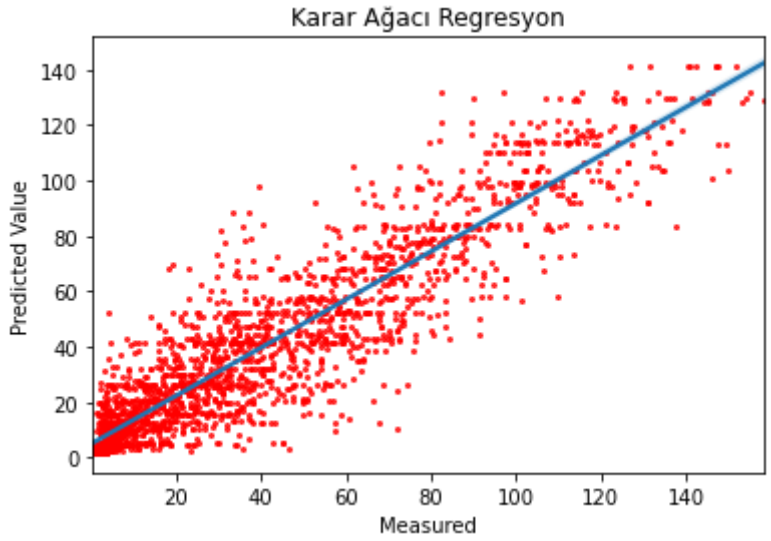
Çizelge 4.8. Kültürpark istasyonu için KA regresyonda GridSearchCV kullanarak en iyi parametre değerlerini ayarlama

Parametreler	Değerler	En iyi değer
max_depth	[5, 8, 10]	10
min_samples_leaf	[2, 10, 80, 100]	15
min_samples_split	[10, 15, 20]	2

KA regresyon modellemesi kullanarak gözlenen değerlere karşı tahmin edilen O_3 konsantrasyon değerlerinin grafikleri Şekil 4.11 ve Şekil 4.12'de gösterilmektedir.



Şekil 4.11. Kültürpark istasyonu için 1-13 Kasım tarihleri arasında KA regresyon kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerin karşılaştırılması grafiği



Şekil 4.12. Kültürpark istasyonu için KA regresyon kullanarak tahmin edilen saatlik O_3 değerlerin dağılım grafiği

4.2.2. Rastgele orman regresyon sonuçları

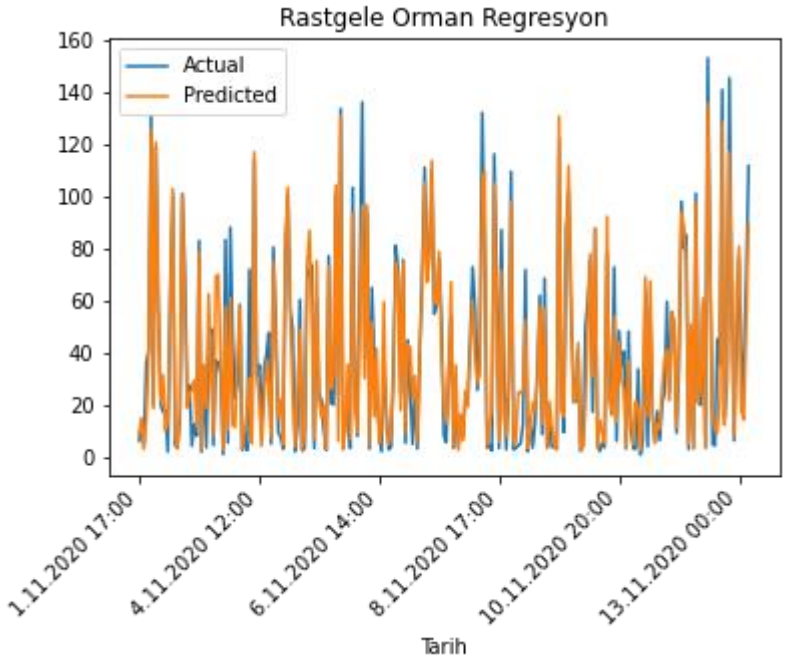
RO regresyon modellemesinin optimum değerleri Python'daki ızgara arama (Grid Search) işlevi kullanılarak beş kat çapraz doğrulama tekniği ile elde edildi. RO algoritması için kullanılan parametre değerleri ve optimum sonuçlar Çizelge 4.9'da gösterilmiştir. Model

için en iyi kombinasyonun max_depth parametresinin 10, min_samples_leaf parametresinin 10, min_samples_split parametresinin 2 olduğu durumda bulunmuştur. Test seti için bulunan hata metrikleri Çizelge 4.7'de özetlenmiştir. KOKH değeri 11,816, OKH değeri 139,617, OMH değeri 8,443, OMYH değeri 0,559 ve R^2 değeri 0,891 bulunmuştur.

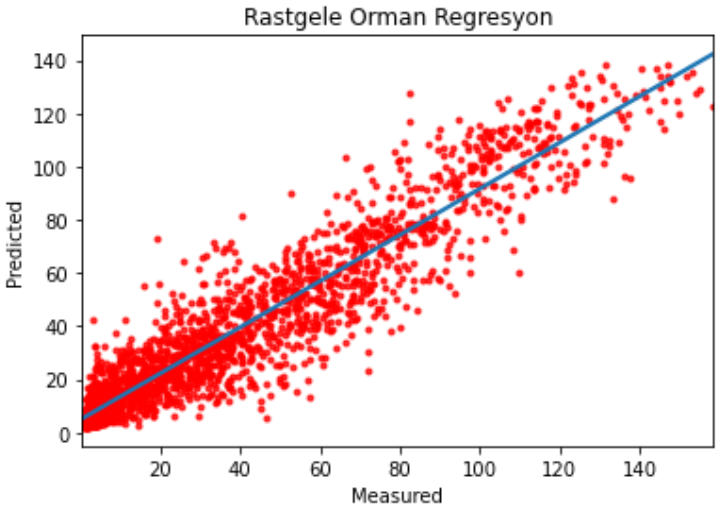
Çizelge 4.9. Kültürpark istasyonu için RO regresyonda GridSearchCV kullanarak en iyi parametre değerlerini ayarlama

Parametreler	Değerler	En iyi değer
max_depth	[5, 8, 10]	10
min_samples_leaf	[2, 10, 80, 100]	10
min_samples_split	[10, 15, 20]	2

RO regresyon modellemesi kullanarak gözlenen değerlere karşı tahmin edilen O_3 konsantrasyon değerlerinin grafikleri Şekil 4.13 ve Şekil 4.14'te gösterilmektedir.



Şekil 4.13. Kültürpark istasyonu için 1-13 Kasım tarihleri arasında RO regresyon kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerin karşılaştırılması grafiği



Şekil 4.14. Kültürpark istasyonu için RO regresyon kullanarak tahmin edilen saatlik O_3 değerlerin dağılım grafiği

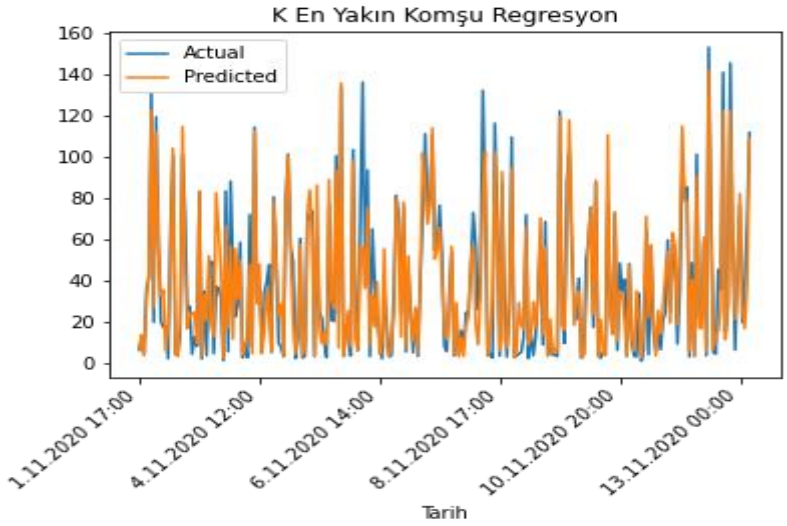
4.2.3. k-en yakın komşu regresyon sonuçları

k-EK modellemesinin optimum değerleri Python'daki ızgara arama (Grid Search) işlevi kullanılarak beş kat çapraz doğrulama tekniği ile elde edildi. Algoritma için kullanılan parametre değerleri ve optimum sonuç Çizelge 4.10'da gösterilmiştir. Model için en iyi sonucun n_neighbors parametresinin 6 olduğu durumda bulunmuştur. Test seti için bulunan hata metrikleri Çizelge 4.7'de özetlenmiştir. KOKH değeri 13,763, OKH değeri 189,422, OMH değeri 9,512, OMYH değeri 0,664 ve R^2 değeri 0,852 bulunmuştur.

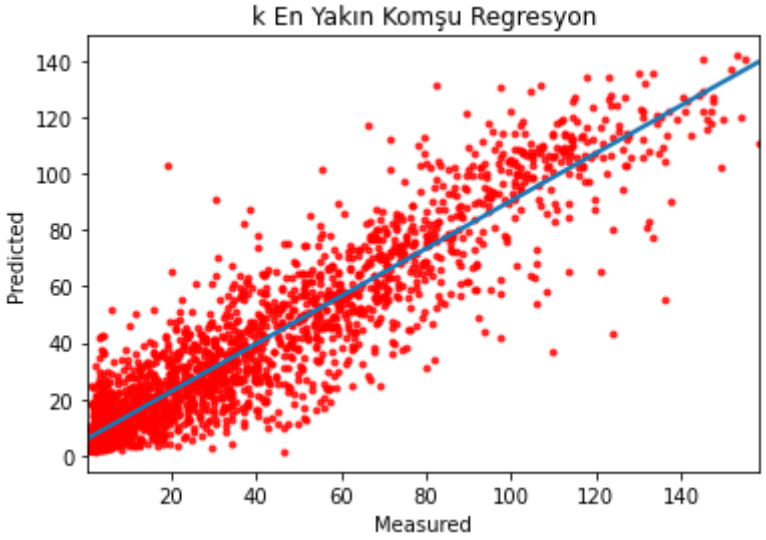
Çizelge 4.10. Kültürpark istasyonu için k-EK regresyonda GridSearchCV kullanarak en iyi parametre değerlerini ayarlama

Parametreler	Değerler	En iyi değer
n_neighbors	(1, 30, 1)	6

k-EK regresyon modellemesi kullanarak gözlenen değerlere karşı tahmin edilen O_3 konsantrasyonunun dağılım grafikleri Şekil 4.15 ve Şekil 4.16'da gösterilmektedir.



Şekil 4.15. Kültürpark istasyonu için 1-13 Kasım tarihleri arasında k-EK kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerin karşılaştırılması grafiği



Şekil 4.16. Kültürpark istasyonu için k-EK kullanarak tahmin edilen saatlik O_3 değerlerin dağılım grafiği

4.2.4. Çok katmanlı algılayıcı regresyon sonuçları

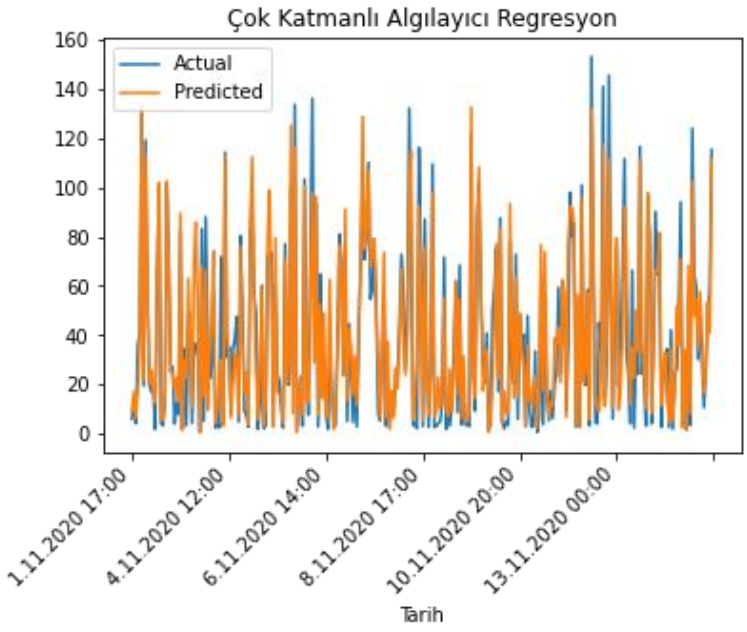
ÇKA regresyon modellemesinin optimum değerleri Python'daki ızgara arama (Grid Search) işlevi kullanılarak beş kat çapraz doğrulama tekniği ile elde edildi. Algoritma için kullanılan parametre değerleri ve optimum sonuçlar Çizelge 4.11'de gösterilmiştir. Model için en iyi kombinasyonun alpha parametresinin 0,1, hidden_layer_sizes parametresinin (100, 100) olduğu durumda bulunmuştur. Test seti için bulunan hata

metrikleri Çizelge 4.7'de özetlenmiştir. KOKH değeri 13,201, OKH değeri 174,257, OMH değeri 9,606, OMYH değeri 0,654 ve R^2 değeri 0,864 bulunmuştur.

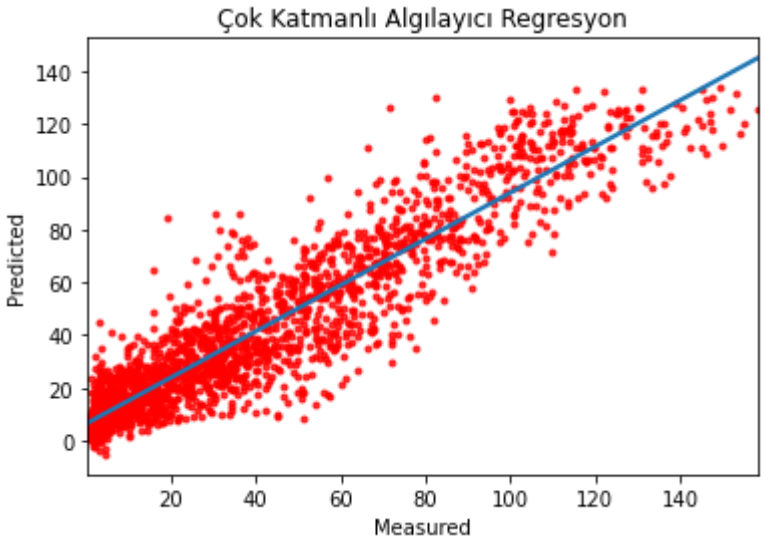
Çizelge 4.11. Kültürpark istasyonunda ÇKA regresyon için GridSearchCV kullanarak en iyi parametre değerlerini ayarlama

Parametreler	Değerler	En iyi değer
alpha	[0,1, 0,01, 0,02, 0,001, 0,0001]	0,1
hidden_layer_sizes	[(10,20), (5,5), (100,100)]	(100, 100)

ÇKA regresyon modellemesi kullanarak gözlenen değerlere karşı tahmin edilen O_3 konsantrasyonunun grafikleri Şekil 4.17 ve Şekil 4.18'te gösterilmektedir.



Şekil 4.17. Kültürpark istasyonu için 1-13 Kasım tarihleri arasında ÇKA regresyon kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerin karşılaştırılması grafiği



Şekil 4.18. Kültürpark istasyonu için ÇKA regresyon kullanarak tahmin edilen saatlik O_3 değerlerin dağılım grafiği

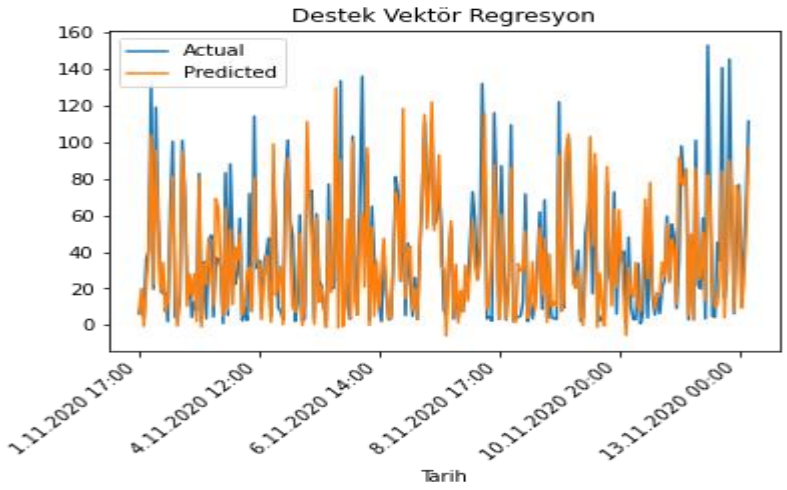
4.2.5. Destek vektör makinesi regresyon sonuçları

DVR modellemesinin optimum değerleri Python'daki ızgara arama (Grid Search) işlevi kullanılarak beş kat çapraz doğrulama tekniği ile elde edildi. Algoritma için kullanılan parametre değeri ve optimum sonuç Çizelge 4.12'de gösterilmiştir. Deneysel sonuçlar, DVR modeli için en iyi kombinasyonun C parametresinin 1000 değer aldığı durumda göstermiştir. Test seti için bulunan hata metrikleri Çizelge 4.7'de özetlenmiştir. KOKH değeri 17,878, OKH değeri 319,608, OMH değeri 12,886, OMYH değeri 0,957 ve R^2 değeri 0,751 bulunmuştur.

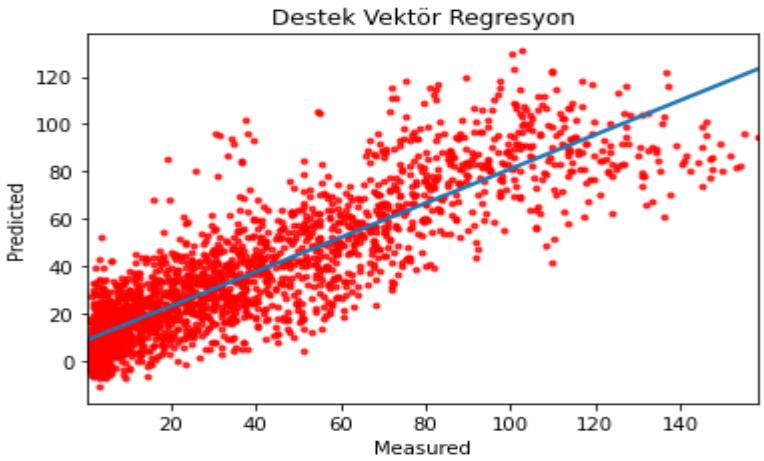
Çizelge 4.12. Kültürpark istasyonu için DVR algoritmasında GridSearchCV kullanarak en iyi parametre değerlerini ayarlama

Parametreler	Değerler	En iyi değer
C	[0,1, 1, 10, 100, 1000]	1000

DVR modellemesi kullanarak gözlenen değerlere karşı tahmin edilen O_3 konsantrasyonunun grafikleri Şekil 4.19 ve Şekil 4.20'de gösterilmektedir.



Şekil 4.19. Kültürpark istasyonu için 1-13 Kasım tarihleri arasında DVR kullanarak gerçek ve tahmin edilen saatlik O_3 değerlerin karşılaştırılması grafiği



Şekil 4.20. Kültürpark istasyonu için DVR kullanarak tahmin edilen saatlik O_3 değerlerin dağılım grafiği

5. SONUÇ

Hava kalitesini tahmin etmek, kirleticilerin ve partiküllerin dinamik yapısı, uçuculuğu ve uzay ve zamandaki yüksek değişkenliği nedeniyle karmaşık bir iştir. Aynı zamanda, özellikle kentsel alanlarda, hava kirliliğinin nüfus ve çevre için gözlemlenen kritik etkileri nedeniyle hava kalitesini modelleyebilmek, tahmin edebilmek ve izleyebilmek giderek daha önemli hale gelmektedir.

Bu çalışmada, O_3 konsantrasyonlarını tahmin etmek için makine öğrenimi algoritmalarına dayalı regresyon modelleri oluşturulmuştur. Kullanılan makine öğrenmesi regresyon algoritmaları; RO, KA, DVM, k-EK ve ÇKA regresyonudur.

Beş yöntemin O_3 konsantrasyonlarının değerlerini tahmin etmedeki performansı KOKH, OKH, OMH ve OMYH, R^2 olmak üzere beş kritere göre değerlendirilmiştir. RO regresyonu düşük hata değerleri ve yüksek R^2 ile kullanılan diğer dört algoritma arasından en iyi yöntem olduğu sonucuna varılmıştır. DVM, iki istasyon için kullanılan veri setlerinde en yüksek işleme süresini elde ettiği, daha yüksek bir hata oranı verdiği ve daha düşük açıklama katsayısına ulaştığı için tüm algoritmaların en kötüsünü gerçekleştirmiştir.

Gelecekteki çalışmalarda, ozon tahmini için RNN ve LSTM gibi derin öğrenme teknikleri de dikkate alınabilir. Ayrıca, makine öğrenimi algoritmaları tarafından elde edilen sonuçlar derin öğrenme yöntemleri kullanarak elde edilen sonuçlarla karşılaştırılabilir.

KAYNAKLAR

Abirami, S., ve Chitra, P. (2020). The Digital Twin Paradigm for Smarter Systems and Environments: The Industry Use Cases. *Advances in Computers*, *117*(1), 339-368. https://doi.org/10.1016/bs.adcom.2019.09.007

Achieng, K. O. (2019). Modelling of soil moisture retention curve using machine learning techniques: Artificial and deep neural networks vs support vector regression models. *Computers and Geosciences*, *133*, 1-17. https://doi.org/10.1016/j.cageo.2019.104320

Aljanabi, M., Shkoukani, M., ve Hijjawi, M. (2020). Ground-level Ozone Prediction Using Machine Learning Techniques: A Case Study in Amman, Jordan. *International Journal of Automation and Computing*, *17*(5), 667–677. https://doi.org/10.1007/s11633-020-1233-4

Ishak, A. B., Daoud, M. B., ve Trabelsi, A. (2017). Ozone Concentration Forecasting Using Statistical Learning Approaches. *Journal of Materials and Environmental Sciences*, *8*(12), 4532–4543. https://doi.org/10.26872/jmes.2017.8.12.478

Bozdağ, A., Dokuz, Y., ve Gökçek, Ö. B. (2020). Spatial prediction of PM10 concentration using machine learning algorithms in Ankara, Turkey. *Environmental Pollution*, *263*,114635. https://doi.org/10.1016/j.envpol.2020.114635

Capilla, C. (2016). Prediction of hourly ozone concentrations with multiple regression and multilayer perceptron models. *International Journal of Sustainable Development and Planning*, *11*(4). https://doi.org/10.2495/SDP-V11-N4-558-565

Castelli, M., Clemente, F. M., Popovič, A., Silva, S., ve Vanneschi, L. (2020). A Machine Learning Approach to Predict Air Quality in California. *Complexity*, *2020*, 1–23. https://doi.org/10.1155/2020/8049504

Chattopadhyay, G., Midya, S. K., ve Chattopadhyay, S. (2019). MLP based predictive model for surface ozone concentration over an urban area in the Gangetic West Bengal during pre-monsoon season. *Journal of Atmospheric and Solar-Terrestrial Physics*, *184*, 57–62. https://doi.org/10.1016/J.JASTP.2019.01.008

Chattopadhyay, S., ve Chattopadhyay, G. (2012). Modeling and Prediction of Monthly Total Ozone Concentrations by Use of an Artificial Neural Network Based on Principal Component Analysis. *Pure and Applied Geophysics*, *169*(10), 1891-1908. https://doi.org/10.1007/s00024-011-0437-5

Chelani, A. B. (2010). Prediction of daily maximum ground ozone concentration using support vector machine. *Environmental Monitoring and Assessment*, *162*(1–4), 169–176. https://doi.org/10.1007/s10661-009-0785-0

Choubin, B., Abdolshahnejad, M., Moradi, E., Querol, X., Mosavi, A., Shamshirband, S., ve Ghamisi, P. (2020). Spatial hazard assessment of the PM10 using machine learning

models in Barcelona, Spain. *Science of the Total Environment, 701, 134474*. https://doi.org/10.1016/j.scitotenv.2019.134474

Cortes C, ve Vapnik V. (1995). Support-vector networks. *Machine Learning, 20*, 273–297.

Çevre, Şehircilik ve İklim Değişikliği Bakanlığı. (2022, Ocak 9). Ulusal Hava Kalitesi İzleme Ağı. http://index.havaizleme.gov.tr/Report/Station

Díaz-Robles, L. A., Ortega, J. C., Fu, J. S., Reed, G. D., Chow, J. C., Watson, J. G., ve Moncada-Herrera, J. A. (2008). A hybrid ARIMA and artificial neural networks model to forecast particulate matter in urban areas: The case of Temuco, Chile. *Atmospheric Environment, 42*(35), 8331-8340. https://doi.org/10.1016/j.atmosenv.2008.07.020

Doreswamy, Harishkumar, K. S., Km, Y., ve Gad, I. (2020). Forecasting Air Pollution Particulate Matter (PM2.5) Using Machine Learning Regression Models. *Procedia Computer Science, 171, 2057-2066*. https://doi.org/10.1016/j.procs.2020.04.221

Faleh, R., Bedoui, S., ve Kachouri, A. (2017). Ozone monitoring using support vector machine and K-nearest neighbors methods. *Journal of Electrical and Electronics Engineering, 10*(1), 49-52.

Frouin, A., Dandine-Roulland, C., Pierre-Jean, M., Deleuze, J. F., Ambroise, C., ve le Floch, E. (2020). Exploring the Link Between Additive Heritability and Prediction Accuracy From a Ridge Regression Perspective. *Frontiers in Genetics, 11, 1-15*. https://doi.org/10.3389/fgene.2020.581594

Guabassi, I. el, Bousalem, Z., Marah, R., ve Qazdar, A. (2021). A Recommender System for Predicting Students' Admission to a Graduate Program using Machine Learning Algorithms. *International Journal of Online and Biomedical Engineering, 17*(2), 135-147. https://doi.org/10.3991/ijoe.v17i02.20049

Hastie, T., Tibshirani, R., ve Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. *The Mathematical Intelligencer, 27*(2), 83-85

Jain, A. K., Mao, J., ve Mohiuddin, K. M. (1996). Artificial neural networks: a tutorial. *Computer, 29*(3), 31-44. https://doi.org/10.1109/2.485891

Jumin, E., Zaini, N., Ahmed, A. N., Abdullah, S., Ismail, M., Sherif, M., Sefelnasr, A., ve El-Shafie, A. (2020). Machine learning versus linear regression modelling approach for accurate ozone concentrations prediction. *Engineering Applications of Computational Fluid Mechanics, 14*(1), 713–725. https://doi.org/10.1080/19942060.2020.1758792

Kotsiantis, S. B., Kanellopoulos, D., ve Pintelas, P. (2006). Data preprocessing for supervised leaning. *International Journal of Computer Science, 1*(2), 111-117. https://doi.org/10.1080/02331931003692557

Krishnan, S. (2021). Biomedical Signal Analysis for Connected Healthcare. *Elsevier*, https://doi.org/10.1016/b978-0-12-813086-5.00006-2

Kumar, V., ve Sahu, M. (2021). Evaluation of nine machine learning regression algorithms for calibration of low-cost PM2.5 sensor. *Journal of Aerosol Science*, *157*, 105809. https://doi.org/10.1016/j.jaerosci.2021.105809

Li, Z., Yim, S. H. L., ve Ho, K. F. (2020). High temporal resolution prediction of street-level PM2.5 and NOx concentrations using machine learning approach. *Journal of Cleaner Production*, *268*, 1-10. https://doi.org/10.1016/j.jclepro.2020.121975

Liang, Y. C., Maimury, Y., Chen, A. H. L., ve Juarez, J. R. C. (2020). Machine learning-based prediction of air quality. *Applied Sciences (Switzerland)*, *10*(24), 1-17. https://doi.org/10.3390/app10249151

Lin, S. W., Ying, K. C., Lee, C. Y., ve Lee, Z. J. (2012). An intelligent algorithm with feature selection and decision rules applied to anomaly intrusion detection. *Applied Soft Computing Journal*, *12*(10), 3285-3290. https://doi.org/10.1016/j.asoc.2012.05.004

Liu, H., Li, Q., Yu, D., ve Gu, Y. (2019). Air quality index and air pollutant concentration prediction based on machine learning algorithms. *Applied Sciences (Switzerland)*, *9*(19), 1-9. https://doi.org/10.3390/app9194069

Mahmood, M. A., Visan, A. I., Ristoscu, C., ve Mihailescu, I. N. (2021). Artificial neural network algorithms for 3D printing. *Materials*, *14*(1), 163. https://doi.org/10.3390/ma14010163

Maleki, H., Sorooshian, A., Goudarzi, G., Baboli, Z., Tahmasebi Birgani, Y., ve Rahmati, M. (2019). Air pollution prediction by using an artificial neural network model. *Clean Technologies and Environmental Policy*, *21*(6), 1341-1352. https://doi.org/10.1007/s10098-019-01709-w

Melkumova, L. E., ve Shatskikh, S. Y. (2017). Comparing Ridge and LASSO estimators for data analysis. *Procedia Engineering*, *201*. https://doi.org/10.1016/j.proeng.2017.09.615

Nagalla, R., Pothuganti, P., ve Pawar, D. S. (2017). Analyzing Gap Acceptance Behavior at Unsignalized Intersections Using Support Vector Machines, Decision Tree and Random Forests. *Procedia Computer Science*, *109*, *474-481*. https://doi.org/10.1016/j.procs.2017.05.312

Ortiz-García, E. G., Salcedo-Sanz, S., Pérez-Bellido, Á. M., Portilla-Figueras, J. A., ve Prieto, L. (2010). Prediction of hourly O3 concentrations using support vector regression algorithms. *Atmospheric Environment*, *44*(35), 4481-4488. https://doi.org/10.1016/j.atmosenv.2010.07.024

Osarogiagbon, A. U., Khan, F., Venkatesan, R., ve Gillard, P. (2021). Review and analysis of supervised machine learning algorithms for hazardous events in drilling operations. In *Process Safety and Environmental Protection*, *147*, 367-384. https://doi.org/10.1016/j.psep.2020.09.038

Pattnaik, P., Sharma, A., Choudhary, M., Singh, V., Agarwal, P., ve Kukshal, V. (2020). Role of machine learning in the field of Fiber reinforced polymer composites: A preliminary discussion. *Materials Today: Proceedings*, *44*(6), 4703-4708. https://doi.org/10.1016/j.matpr.2020.11.026

Prieto, L. (2010). Prediction of hourly O3 concentrations using support vector regression algorithms. *Atmospheric Environment*, *44*(35), 4481-4488. https://doi.org/10.1016/j.atmosenv.2010.07.024

Rajab, J. M., MatJafri, M. Z., ve Lim, H. S. (2013). Combining multiple regression and principal component analysis for accurate predictions for column ozone in Peninsular Malaysia. *Atmospheric Environment*, *71*, 36-43. https://doi.org/10.1016/j.atmosenv.2013.01.019

Ribeiro, V. M. (2021). Sulfur dioxide emissions in Portugal: Prediction, estimation and air quality regulation using machine learning. *Journal of Cleaner Production*, *317*, 128358. https://doi.org/10.1016/j.jclepro.2021.128358

Sahin, F., Kara, M. K., Koc, A., ve Sahin, G. (2020). Multi-criteria decision-making using GIS-AHP for air pollution problem in Igdir Province/Turkey. *Environmental Science and Pollution Research, 27*(29), 36215–36230. https://doi.org/10.1007/s11356-020-09710-3

SEPA (Scottish Environment Protection Agency). (2022, Ocak 26). The chemistry of air pollution. https://www.sepa.org.uk/media/120465/mtc_chem_of_air_pollution.pdf

Shobha, G., Rangaswamy, S. (2018). Computational Analysis and Understanding of Natural Languages: Principles, Methods and Applications. *Handbook of Statistics, 38*, 197-228. https://doi.org/10.1016/bs.host.2018.07.004.

Smola, A. J., ve Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and Computing*, *14*(3), 199-222. https://doi.org/10.1023/B:STCO.0000035301.49549.88

Sousa, S. I. V., Martins, F. G., Alvim-Ferraz, M. C. M., ve Pereira, M. C. (2007). Multiple linear regression and artificial neural networks based on principal components to predict ozone concentrations. *Environmental Modelling and Software, 22*(1), 97-103. https://doi.org/10.1016/j.envsoft.2005.12.002

Stafoggia, M., Bellander, T., Bucci, S., Davoli, M., de Hoogh, K., de' Donato, F., Gariazzo, C., Lyapustin, A., Michelozzi, P., Renzi, M., Scortichini, M., Shtein, A., Viegi, G., Kloog, I., ve Schwartz, J. (2019). Estimation of daily PM10 and PM2.5 concentrations in Italy, 2013–2015, using a spatiotemporal land-use random-forest model. *Environment International*, *124*, 170-179. https://doi.org/10.1016/j.envint.2019.01.016

Suárez Sánchez, A., García Nieto, P. J., Riesgo Fernández, P., del Coz Díaz, J. J., ve Iglesias-Rodríguez, F. J. (2011). Application of an SVM-based regression model to the air quality study at local scale in the Avilés urban area (Spain). *Mathematical and Computer Modelling*, *54*(5–6), 1453-1466. https://doi.org/10.1016/j.mcm.2011.04.017

Taoufik, N., Boumya, W., Achak, M., Chennouk, H., Dewil, R., ve Barka, N. (2021). The state of art on the prediction of efficiency and modeling of the processes of pollutants removal based on machine learning. *Science of the Total Environment, 807*(2022), 1-16. https://doi.org/10.1016/j.scitotenv.2021.150554

Tibshirani, R. (2011). Regression shrinkage and selection via the lasso: A retrospective. *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, *73*(3), 273-282. https://doi.org/10.1111/j.1467-9868.2011.00771.x

Uyanık, T., Karatuğ, Ç., ve Arslanoğlu, Y. (2020). Machine learning approach to ship fuel consumption: A case of container vessel. *Transportation Research Part D: Transport and Environment, 84*, 1-14. https://doi.org/10.1016/j.trd.2020.102389

Vakharia, V., Castelli, I. E., Bhavsar, K., ve Solanki, A. (2021). Bandgap prediction of metal halide perovskites using regression machine learning models. *Physics Letters A*, *422*, 127800. https://doi.org/10.1016/J.PHYSLETA.2021.127800

Vapnik, V. N. (2000). The Nature of Statistical Learning Theory. In *The Nature of Statistical Learning Theory* (s.1-15). https://doi.org/10.1007/978-1-4757-3264-1

Wang, W., ve Xu, Z. (2004). A heuristic training for support vector regression. *Neurocomputing*, 61, 259–275. https://doi.org/10.1016/j.neucom.2003.11.012

Yafouz, A., AlDahoul, N., Birima, A. H., Ahmed, A. N., Sherif, M., Sefelnasr, A., Allawi, M. F., ve Elshafie, A. (2021). Comprehensive comparison of various machine learning algorithms for short-term ozone concentration prediction. *Alexandria Engineering Journal*. https://doi.org/10.1016/j.aej.2021.10.021

EKLER

EK 1 Veri kümelerinin detayları

EK 2 Bursa Uludağ Üniversitesi istasyonu için model geliştirilmesi ve performans değerlendirmesi Python kodları

EK 1

Çizelge EK 1.1. Kültürpark istasyonu veri kümesi için öznitelik özellikleri

#	Column	Non-Null Count	Dtype
0	Tarih	47393 non-null	object
1	PM2.5	16406 non-null	float64
2	SO2	46678 non-null	float64
3	NO2	44575 non-null	float64
4	NO	43633 non-null	float64
5	NOX	44369 non-null	float64
6	O3	46305 non-null	float64
7	Hava Sıcaklığı	46838 non-null	float64
8	Rüzgar Yönü	35002 non-null	float64
9	Rüzgar Hızı	46611 non-null	float64
10	Bağıl Nem	46764 non-null	float64
11	Hava Basıncı	46927 non-null	float64

Çizelge EK 1.2. Kültürpark istasyonu veri kümesinde özniteliklerin boş değer sayısı

Tarih	0
PM2.5	30987
SO2	715
NO2	2818
NO	3760
NOX	3024
O3	1088
Hava Sıcaklığı	555
Rüzgar Yönü	12391
Rüzgar Hızı	782
Bağıl Nem	629
Hava Basıncı	466

Çizelge EK 1.3. Bursa Uludağ Üniversitesi istasyonu veri kümesi için öznitelik özellikleri

#	Column	Non-Null Count	Dtype
0	Tarih	47393 non-null	object
1	PM2.5	46794 non-null	float64
2	SO2	44686 non-null	float64
3	NO2	41403 non-null	float64
4	NO	37698 non-null	float64
5	NOX	45013 non-null	float64
6	O3	45233 non-null	float64
7	Hava Sıcaklığı	46632 non-null	float64
8	Rüzgar Yönü	42885 non-null	float64
9	Rüzgar Hızı	46850 non-null	float64
10	Bağıl Nem	46736 non-null	float64
11	Hava Basıncı	46854 non-null	float64

Çizelge EK 1.4. Bursa Uludağ Üniversitesi istasyonu veri kümesinde özniteliklerin boş değer sayısı

Tarih	0
PM2.5	599
SO2	2707
NO2	5990
NO	9695
NOX	2380
O3	2160
Hava Sıcaklığı	761
Rüzgar Yönü	4508
Rüzgar Hızı	543
Bağıl Nem	657
Hava Basıncı	539

EK 2

Bursa Uludağ Üniversitesi istasyonu için model geliştirme ve performans değerlendirme

```
from sklearn.model_selection import train_test_split

x_train, x_test,y_train,y_test = train_test_split(x,y,test_size=0.20, random_state=0)
```

```
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error,mean_absolute_error,r2_score,mean_absolute_error,mean_absolute_percentage_erro
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.feature_selection import RFE
from sklearn.model_selection import GridSearchCV
```

```
from sklearn.tree import DecisionTreeRegressor
dtr=DecisionTreeRegressor()
```

```
dtr_params={"max_depth":[5,8,10],
           "min_samples_split":[2,10,80,100],
           "min_samples_leaf":[10,15,20]}
```

```
dtr_cv_model=GridSearchCV(dtr, dtr_params,cv=5, n_jobs=-1, verbose=2).fit(x_train,y_train)
```

```
dtr_cv_model.best_params_
```

```
dtr=DecisionTreeRegressor(max_depth=10,min_samples_split=2,min_samples_leaf=20).fit(x_train,y_train)
y_pred_dtr=dtr.predict(x_test)
print('RMSE of DTR model:%.5f' % np.sqrt(mean_squared_error(y_test,y_pred_dtr)))
```

```
import math
```

```
dframe=pd.DataFrame({'Actual':y_test.flatten(),'Predicted':y_pred_dtr.flatten()})
# dframe=pd.DataFrame({'Actual':y_test,'Predicted':y_pred_dtr})
```

```
dframe.head()
```

	Actual	Predicted
0	68.51	74.127500
1	88.90	107.862381
2	40.38	18.191538
3	52.03	48.123370
4	147.49	125.888427

```
dframe.set_index(veri['Tarih'][19000:24703],inplace=True)
```

```
graph=dframe[5000:5300]
graph
```

```
plt.figure(figsize=(25,100))
graph.plot(kind='line',title="Karar Ağacı Regresyon")
plt.xticks(rotation=45,fontsize='medium',fontweight='light',horizontalalignment='right')
```

```
print('%.6f' % mean_squared_error(y_test,y_pred_dtr))
```

```
print('%.6f' % r2_score(y_test,y_pred_dtr))
```

```
print('%.6f' % mean_absolute_error(y_test,y_pred_dtr))
```

```
import matplotlib.pyplot as plt
plt.xlabel("Measured")
plt.ylabel("Predicted Value")
plt.title("Karar Ağacı Regresyon")
g=sns.regplot(x=y_test,y=y_pred_dtr, scatter_kws={'color':'r','s':4})
plt.show()
```

```
from sklearn.ensemble import RandomForestRegressor
rf=RandomForestRegressor()
```

```
rf_params={"max_depth":[5,8,10],
           "min_samples_split":[2,10,80,100],
           "min_samples_leaf":[10,15,20]}
```

```
rf_cv_model=GridSearchCV(rf, rf_params,cv=5, n_jobs=-1, verbose=2).fit(x_train,y_train)
```

```
rf_cv_model.best_params_
```

```
rf=RandomForestRegressor(max_depth=10,min_samples_split=10,min_samples_leaf=10).fit(x_train,y_train)
```

```
y_pred_rf=rf.predict(x_test)
```

```
print('RMSE of predicted  RH in RF model: %.5f' %  np.sqrt(mean_squared_error(y_test,y_pred_rf)))
```

```
print('%.6f' % mean_squared_error( y_test,y_pred_rf))
```

```
print('%.6f' % r2_score(y_test,y_pred_rf))
```

```
print('%.6f' % mean_absolute_error(y_test,y_pred_rf))
```

```
print('%.6f' % mean_absolute_percentage_error(y_test,y_pred_rf))
```

```
import matplotlib.pyplot as plt

plt.xlabel('Measured')
plt.ylabel("Predicted")
plt.title("Rastgele Orman Regresyon")
g=sns.regplot(x=y_test,y=y_pred_rf, ci=None, scatter_kws={'color':'r','s':9})
plt.show()
```

```
dframe_rf=pd.DataFrame({'Actual':y_test.flatten(),'Predicted':y_pred_rf.flatten()})
```

```
dframe_rf.set_index(veri['Tarih'][19000:24703],inplace=True)
```

```
graph_rf=dframe_rf[5000:5300]
graph_rf
```

```
graph_rf.plot(kind='line',title="Rastgele Orman Regresyon")
plt.xticks(rotation=45,fontsize='medium',fontweight='light',horizontalalignment='right')
```

```
from sklearn.svm import SVR
```

```
sv_reg=SVR()
```

```
svr_params={'C':[0.1,1,10,100,1000]}
```

```
sv_cv_model=GridSearchCV(sv_reg,svr_params,cv=5,verbose=2,n_jobs=-1).fit(x_train,y_train)
```

```
sv_cv_model.best_params_
```

```
sv_model=SVR(C=1000).fit(x_train,y_train)
y_pred_sv=sv_model.predict(x_test)
print('RMSE of SVR model:%.3f' % np.sqrt(mean_squared_error(y_test,y_pred_sv)))
```

```
print('%.6f' % mean_squared_error(y_test,y_pred_sv))
```

```
print('%.6f' % r2_score(y_test,y_pred_sv))
```

```
print('%.6f' % mean_absolute_error(y_test,y_pred_sv))
```

```
print('%.6f' % mean_absolute_percentage_error(y_test,y_pred_sv))
```

```
import matplotlib.pyplot as plt

plt.xlabel('Measured')
plt.ylabel("Predicted")
plt.title("Destek Vektör Makinesi Regresyonu")

g=sns.regplot(x=y_test,y=y_pred_sv,data=veri, ci=None, scatter_kws={'color':'r','s':9})
plt.show()
```

```
dframe_sv=pd.DataFrame({'Actual':y_test.flatten(),'Predicted':y_pred_sv.flatten()})
```

```
dframe_sv.set_index(veri['Tarih'][19000:24703],inplace=True)
```

```
graph_sv=dframe_sv[5000:5300]
graph_sv
```

```
graph_sv.plot(kind='line',title="Destek Vektör Makinesi Regresyonu")
plt.xticks(rotation=45,fontsize='medium',fontweight='light',horizontalalignment='right')
```

```
from sklearn.linear_model import Ridge
from sklearn.linear_model import Lasso
from sklearn.linear_model import ElasticNet
from sklearn.neighbors import KNeighborsRegressor
from sklearn.neural_network import MLPRegressor
from sklearn.linear_model import RidgeCV
from sklearn import model_selection
```

```
knn_params={"n_neighbors":np.arange(1,30,1)}
```

```
knn_cv_model=GridSearchCV(kn_reg,knn_params,cv=5,verbose=2,n_jobs=-1).fit(x_train,y_train)
```

```
knn_cv_model.best_params_
```

```
kn_model=KNeighborsRegressor(n_neighbors=10).fit(x_train,y_train)
```

```
y_pred_kn=kn_model.predict(x_test)
print('RMSE of knn model:%.3f' % np.sqrt(mean_squared_error(y_test,y_pred_kn)))
```

```
print('%.5f' % mean_squared_error(y_test,y_pred_kn))
```

```
print('%.5f' % r2_score(y_test,y_pred_kn))
```

```
print('%.6f' % mean_absolute_error(y_test,y_pred_kn))
```

```
print('%.6f' % mean_absolute_percentage_error(y_test,y_pred_kn))
```

```
import matplotlib.pyplot as plt
plt.xlabel("Measured")
plt.ylabel("Predicted")
plt.title("k En Yakın Komşu Regresyon")
g=sns.regplot(x=y_test,y=y_pred_kn,data=veri, ci=None, scatter_kws={'color':'r','s':9})
plt.show()
```

```
dframe_kn=pd.DataFrame({'Actual':y_test.flatten(),'Predicted':y_pred_kn.flatten()})
```

```
dframe_kn.set_index(veri['Tarih'][19000:24703],inplace=True)
```

```
graph_kn=dframe_kn[5000:5300]
graph_kn
```

```
graph_kn=dframe_kn.head(300)
```

```
graph_kn.plot(kind='line',title="k En Yakın Komşu Regresyon")
plt.xticks(rotation=45,fontsize='medium',fontweight='light',horizontalalignment='right')
```

```
mlp_params={"alpha":[0.1,0.01,0.02,0.001,0.0001],
           "hidden_layer_sizes":[(10,20),(5,5),(100,100)]}
```

```
mlp_cv_model=GridSearchCV(mlp,mlp_params,cv=5,verbose=2,n_jobs=-1).fit(x_train,y_train)
```

```
mlp_cv_model.best_params_
```

```
{'alpha': 0.0001, 'hidden_layer_sizes': (100, 100)}
```

```
mlp_model=MLPRegressor(alpha=0.001,hidden_layer_sizes=(100, 100)).fit(x_train,y_train)
y_pred_mlp=mlp_model.predict(x_test)
print('RMSE of SVR model:%.3f' % np.sqrt(mean_squared_error(y_test,y_pred_mlp)))
```

```
print('%.6f' % mean_squared_error(y_test,y_pred_mlp))
```

```
print('%.5f' % r2_score(y_test,y_pred_mlp))
```

```
print('%.6f' % mean_absolute_error(y_test,y_pred_mlp))
```

```
print('%.6f' % mean_absolute_percentage_error(y_test,y_pred_mlp))
```

```
import matplotlib.pyplot as plt
plt.xlabel('Measured')
plt.ylabel("Predicted")
plt.title("Çok Katmanlı Algılayıcı Regresyon")
g=sns.regplot(x=y_test,y=y_pred_mlp,data=veri, ci=None, scatter_kws={'color':'r','s':9})
plt.show()
```

```
dframe_mlp=pd.DataFrame({'Actual':y_test.flatten(),'Predicted':y_pred_mlp.flatten()})
dframe_mlp
```

```
dframe_mlp.set_index(veri['Tarih'][19000:24703],inplace=True)
```

```
graph_mlp=dframe_mlp[5000:5300]
graph_mlp
```

```
graph_mlp.plot(kind='line',title="Çok Katmanlı Algılayıcı Regresyon")
plt.xticks(rotation=45,fontsize='medium',fontweight='light',horizontalalignment='right')
```

ÖZGEÇMİŞ

Adı Soyadı : Ayça Güven
Doğum Yeri ve Tarihi : BURSA-10.12.1993
Yabancı Dil : İngilizce

Eğitim Durumu
Lise : Bursa Cumhuriyet Lisesi
Lisans : Bursa Uludağ Üniversitesi-Endüstri Mühendisliği

Çalıştığı Kurum/Kurumlar : Oyak Renault Otomobil Fabrikaları A.Ş

İletişim (e-posta) : aycaguven5@gmail.com

Yayınları : Güven A., Yağmahan B. (2021) Makine Öğrenmesi Yöntemleri ile Hava Kirliliği Tahmini, *YA/EM'2021-Yöneylem Araştırması/Endüstri Mühendisliği 40. Ulusal Kongresi*, Boğaziçi Üniversitesi, İstanbul, Türkiye, 05 - 07 Temmuz, ss.62.