
GAZİ ÜNİVERSİTESİ
Gazi
1926

MAKİNE ÖĞRENİMİ YÖNTEMLERİ İLE TÜRKİYE İSTATİSTİKİ BÖLGELERİNDE COVID-19 YAYGINLIĞININ ANALİZİ

Hatice Nur CANPOLAT

YÜKSEK LİSANS TEZİ
ENDÜSTRİ MÜHENDİSLİĞİ ANA BİLİM DALI

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

OCAK 2022

ETİK BEYAN

Gazi Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmasında yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Hatice Nur CANPOLAT

31/01/2022

MAKİNE ÖĞRENİMİ YÖNTEMLERİ İLE TÜRKİYE İSTATİSTİKİ BÖLGELERİNDE COVID-19 YAYGINLIĞININ ANALİZİ

(Yüksek Lisans Tezi)

Hatice Nur CANPOLAT

GAZİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

Ocak 2022

ÖZET

Çin'in Wuhan eyaletinde Aralık 2019'da ortaya çıkan COVID-19 salgını, kısa zaman içinde bütün dünyaya yayılmıştır. Dünya üzerindeki COVID-19 hasta sayısının 12 Kasım 2021 tarihi itibariyle 252 milyon 44 bin 337'e ulaştığı, ölümlü vaka sayısının ise 5 milyon 82 bin 431 olduğu görülmektedir. COVID-19 salgınına benzer küresel salgın durumlarında bir ülkenin sağlık sisteminde oluşacak yükün önceden tahmin edilmesi ise hayati derecede önemlidir. Bu çalışmada ise literatürde tahminleme ile ilgili yer alan uygulamalarda etkili performans gösteren rastgele orman regresyonu (RFR), bayes sırt regresyon (BRR) ve lineer regresyon (LR) algoritmaları Türkiye'de COVID-19 salgınının yaygınlığını tahmin etmek amacıyla kullanılmıştır. Çalışmanın literatüre katkısı, Türkiye'nin 12 istatistiki bölgesi COVID-19 vaka sayısını tahmin etmek amacıyla modellerin ayrı ayrı oluşturulmuş olmasıdır. Bölge bazlı vaka sayılarının tahmin edilmesindeki esas amaç farklı iklim ve farklı demografi koşullarının salgınların genel yayılımını etkileyen ana unsurlardan olmasıdır. Ülkeler geneli tahmin modellerinde bölgesel yayılım farklılıklarının oluşturabileceği kısıtlar göz ardı edildiğinden, bölge bazlı tahminlerde daha gerçekçi ve net tahmin modelleri söz konusudur. Elde edilen sonuçlara göre İstanbul bölgesinde R^2=0,98 değeri ile RFR modeli bütün bölgeler için en yüksek tahmin performansını göstermiştir. Aynı bölge için LR modeli R^2=0,69 iken BRR modeli R^2=0,57 değeri ile diğer tahmin modellerinin gerisinde kalmıştır. Bu çalışmanın sonuçları göz önünde bulundurulacak olursa makine öğrenimi temelli regresyon algoritmalarının, çeşitli zaman serisi verileri veya üstel dağılımlı yüksek tahmin güçlüğü olan dalgalı veri setlerinde de etkili analizler gerçekleştirebildiği görülmektedir.

Bilim Kodu	:	90619
Anahtar Kelimeler	:	COVID-19, Türkiye, makine öğrenimi, rastgele orman regresyonu, bayes sırt regresyonu, lineer regresyon
Sayfa Adedi	:	79
Danışman	:	Prof. Dr. Ertan GÜNER

ANALYSIS OF THE PREFERENCE OF COVID-19 IN STATISTICAL REGIONS OF TURKEY BY MACHINE LEARNING METHODS

(M. Sc. Thesis)

Hatice Nur CANPOLAT

GAZİ UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

January 2022

ABSTRACT

The COVID-19 epidemic, which emerged in December 2019 in Wuhan, China, has taken the whole world under its influence in a short time. It is seen that the number of COVID-19 patients in the world has reached 252 million 44 thousand 337 as of 12 November 2021, and the number of fatal cases is 5 million 82 thousand 431. It is vitally important to predict the burden that will occur in a country's health system in global epidemic situations similar to the COVID-19 epidemic. In this study, random forest regression (RFR), Bayesian ridge regression (BRR) and linear regression (LR) algorithms, which show effective performance in the prediction applications in the literature, were used to predict the prevalence of the COVID-19 epidemic in Turkey. The contribution of the study to the literature is that models were created separately to estimate the number of COVID-19 cases in 12 statistical regions of Turkey. The main purpose of estimating the number of cases by region is that different climatic and different demographic conditions are the main factors affecting the general spread of epidemics. Since the constraints that can be created by regional spread differences are ignored in the forecast models across countries, there are more realistic and clear forecast models in region-based forecasts. According to the obtained results, the RFR model is considered the highest estimate for the whole, with the value of R^2=0,98 in the Istanbul region. For the same region, while the LR model was R^2=0,69, the BRR model lagged behind other estimation models with R^2 =0,57. Considering the results of this study, it is seen that machine learning-based regression algorithms can also perform effective analyzes on various time series data or exponentially distributed fluctuating data sets with high estimation difficulty.

Science Code	:	90619
Key Words	:	COVID-19, Turkey, machine learning, random forest regression, bayesian ridge regression, linear regression
Page Number	:	79
Supervisor	:	Prof. Dr. Ertan GÜNER

TEŞEKKÜR

Tez çalışmalarım boyunca kıymetli bilgi, birikim ve tecrübeleri ile bana yol gösterici olan değerli danışman hocam Sayın Prof. Dr. Ertan GÜNER'e teşekkür ve saygılarımı sunarım.

Hayatımın her aşamasında olduğu gibi bu süreçte de bana destek olan aileme çok teşekkür ederim.

İÇİNDEKİLER

Sayfa

Sayfa

ÇİZELGELERİN LİSTESİ

Çizelge **Sayfa**

ŞEKİLLERİN LİSTESİ

Şekil **Sayfa**

HARİTALARIN LİSTESİ

Harita **Sayfa**

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Kısaltmalar	Açıklamalar
BRR	Bayes sırt regresyon
DVM	Destek vektör makinesi
GPS	Küresel konumlandırma sistemi
İBBS	İstatistiki bölge birimleri sınıflandırması
KNN	K-en yakın komşu algoritması
LASSO	En küçük mutlak büzülme ve operatör seçimi
MAE	Ortalama mutlak hata
MSE	Hata kareleri ortalaması
NCA	Düğüm iş birliği algoritması
NSGA-II	Çok amaçlı optimizsayon algoritması
ODE	Adi diferansiyel denklem
RMSE	Kök ortalama kare hata
RFR	Rastgele orman regresyonu
SEIR	Duyarlı-maruz kalan-enfekte-kurtarılmış
SIR	Duyarlı -enfekte-kurtarılmış

1. GİRİŞ

Veri toplama ve depolama teknolojisindeki hızlı gelişmeler, verilerin oluşturulma ve dağıtılma kolaylığı ile birleştiğinde, verinin patlayıcı büyümesini tetikleyerek mevcut büyük veri çağına yol açmıştır. Bu büyük veri kümelerinden eyleme dönüştürülebilir iç görüler elde etmek bilim ve mühendislik; tıp ve biyoteknoloji; hükümet ve bireyler dahil olmak üzere toplumun neredeyse tüm alanlarında karar vermede giderek daha önemli hale gelmektedir. Bununla birlikte veri miktarı (hacim), veri karmaşıklığı (çeşitliliği) ve verinin toplanıp işlenme hızı (hız) insanların yardım almadan analiz edemeyecekleri kadar büyük hale gelmiştir. Bu muazzam veri yığını ve çeşitliliğinin getirdiği zorluklar nedeniyle söz konusu bu büyük veriden faydalı bilgiler çıkarmak için otomatik araçlara ihtiyaç vardır [1].

Makine öğrenimi yöntemleri ise resim, etiketli konuşma, videolar veya nümerik yapılardan oluşan büyük veri kümelerini toplamak ve bu verileri istenen görevi yerine getirerek genel amaçlı öğrenme makinelerini eğitmek için kullanılan otomatik araçlardır. Standart mühendislik süreci, alan bilgisine ve eldeki sorun için optimize edilmiş tasarıma dayanırken makine öğrenimi, büyük miktarda verinin algoritmaları ve çözümleri bulup ortaya çıkarmasına imkân verir. Bu amaçla incelenen kurulumun kesin bir modelini gerektirmek yerine, makine öğrenimi bir hedefin, eğitilecek bir modelin ve bir optimizasyon tekniğinin belirtilmesini gerektirir.

Bir makine öğrenimi yaklaşımı, büyük veri kümesine dayalı olarak bilinen kimyasal reaksiyonların sonucunu tahmin etmek için genel amaçlı bir makineyi eğitecek ve daha sonra da karmaşık moleküller üretmenin yollarını keşfetmek için eğitilmiş algoritmayı kullanarak ilerleyecektir. Benzer bir şekilde, orijinal girdinin bir miktar bozulma ile kurtarılabileceği sıkıştırılmış temsiller elde etmek amacıyla genel amaçlı bir algoritmayı eğitmek için büyük görüntü veya video veri setleri kullanılacaktır.

Bir gerçek hayat probleminin geliştirme maliyeti ve süresi gibi dezavantajları olduğunda veya bazı problemlerin içinde çalışılamayacak kadar karmaşık göründüğü durumlarda, makine öğrenimi geleneksel mühendislik akışına verimli bir alternatif sunabilir. Ancak diğer taraftan, yaklaşımın genel olarak optimumun altında performans sağlama, çözümün yorumlanabilirliğini engelleme veya yalnızca sınırlı bir dizi soruna uygulanma gibi temel dezavantajları da vardır [2].

Teknolojideki gelişmelerin bir sonucu olarak çeşitli endüstri dallarında her gün çok büyük miktarlarda ham veri üretilmektedir. Bu verinin büyük boyutlarda olması ve artan miktarı makine öğrenmesi kavramı üzerine yapılan çalışmalara dikkat çekmektedir. Makine öğrenimi uygulamaları, ilk olarak üretim sistemlerinde oluşan sorunları çözmek için yaklaşık yirmi yıl önce ortaya çıkmıştır. Etkili karar vermeyi destekleyen akıllı sistemler, eşzamanlı üretim hattı programlamak için programlar, makinelerin bakımı için düzenlemeler, üretim görevlerini gerçekleştirmek için makine öğrenimi yöntemlerini kullanan örnekler olarak kabul edilebilir. Makine öğrenimi bilgisayar programlarının deneyimlerine dayalı modelleme ve gelecek olayları gerçeğe en yakın şekilde öngörme konusunda önemli bir yapay zekâ alt dalıdır. Başlıca makine öğrenimi yaklaşımları iki ana kategoride sınıflandırılabilir: denetimli öğrenme ve denetimsiz öğrenme. Denetimli öğrenmede tipik bir sorun sınıflandırmadır, denetimsiz öğrenme ise kümeleme problemlerinde oldukça yaygındır. Sınıflandırma için yaygın olarak kullanılan teknikler arasında yapay sinir ağları, destek vektör makineleri ve karar ağaçları bulunur. Kümeleme tekniği içinde yaygın kullanılan algoritma ise k-ortalamalardır [3].

Çin'in Hubei eyaletinin Wuhan şehrinde 31 Aralık 2019 tarihinde nedeni bilinemeyen zatürre hastaları tespit edilmiş ve 5 Ocak 2020'de Dünya Sağlık Örgütü insanlarda daha önce rastlanmamış yeni bir koronavirüs belirlemiştir [4]. Koronavirüsler (CoV), SARS-CoV ve MERS-CoV virüslerinin de ait olduğu RNA virüsleri grubunda yer alan, insanlarda genel olarak soğuk algınlığına benzer belirtiler göstererek ciddi hastalıklara neden olan virüslerdir [5]. SARS-CoV-2 olarak adlandırılan bu yeni koronavirüsle temasta olan kişilerde, tedavi edilemeyen zatürre benzeri belirtiler gözlemlenmiştir. Başlangıçta 2019-nCoV olarak tanımlanan bu hastalık, daha sonra Covid-19 olarak isimlendirilmiş ve Çin'de tespit edildikten hemen üç ay sonra bütün dünyaya hızla yayılmıştır [6].

12 Mart 2020'de Dünya Sağlık Örgütü'nün küresel salgın olarak ilan ettiği COVID-19 bugün halen bütün insanlığı tehdit etmeye devam etmektedir. Dünya üzerindeki COVID-19 hasta sayısının 12 Kasım 2021 tarihi itibariyle 252 milyon 44 bin 337'e ulaştığı, ölümlü vaka sayısının ise 5 milyon 82 bin 431 olduğu görülmektedir. Türkiye'de ise 11 Mart 2020 tarihinde tespit edilen ilk COVID-19 vakasından 12 Kasım 2021 tarihine kadar ki zaman boyunca toplam hasta sayısı 8 milyon 342 bin 292'ye ulaşmış ve ölümlü vaka sayısının ise 72 bin 910 olduğu bildirilmiştir [7].

Geçmişten günümüze kadar yaşanan salgınların yapısına bakıldığında milyonlarca kişinin ölümüne sebep oldukları görülür. Hatta dünya üzerinde yok olduğu düşünülen bazı salgın hastalıklar kendilerini yenileyerek farklı bölgelerde yeniden ortaya çıkmakta ve insan yaşamını tehdit etmeye devam etmektedir [8].

COVID-19 salgını gelişmiş ülkelerde dahil olmak üzere sağlık alt yapısı, ekonomik sistem, askeri yapı ve sosyal hayat açısından ülkeleri ciddi anlamda olumsuz etkilemiştir. Buradan hareketle salgın hastalıklar ile mücadele konusunda kritik nokta, başlangıç aşamasında hastalığın yayılma potansiyelinin tahmin edilmesi yoluyla gerekli tedbirlerin alınarak salgının yayılmasının önlenmesidir [9].

Buradan hareketle COVID-19 salgınının Türkiye'deki yayılımının incelenmesi ülkemizin etkin sağlık politikaları geliştirebilmesi açısından büyük önem taşımaktadır. Salgın yayılımının Türkiye'nin istatistiki bölgeler bazında ayrı ayrı ele alınması ülkemiz açısından daha etkili bir analiz olacaktır.

Bu çalışmada, Türkiye geneli ve istatistiki 12 bölgesinin ayrı ayrı COVID-19 günlük vaka sayısı dikkate alınarak gelecekteki vaka sayılarını tahmin etmek amacıyla makine öğrenimi algoritmalarından yararlanılmıştır. Her bir istatistiki bölgenin vaka sayılarının tahmin edilmesi hedeflenerek, analiz için makine öğrenimi algoritmalarından rastgele orman regresyon (RFR), bayes sırt regresyon (BRR) ve lineer regresyon (LR) yöntemleri kullanılmıştır. Çalışmanın sonunda kullanılan yöntemler tahmin performansları açısından karşılaştırılmış ve sonuçlar değerlendirilmiştir.

Çalışmanın ikinci bölümünde literatürde yer alan COVID-19 yayınlığı ile ilgili yapılan tahmin çalışmaları detaylı olarak incelenerek özetlenmiştir. Literatür araştırması bölümünde makine öğrenimi algoritmaları kullanılarak yapılan tahmin çalışmalarına yer verilmiş ve mevcut çalışmanın diğer çalışmalardan nasıl farklılaştığı açıklanmıştır. Üçüncü bölüm, çalışmada COVID-19 yaygınlığını tahmin etmek için kullanılan RFR, BRR ve LR yöntemleri ile model performans değerlendirme ölçütleri detaylı olarak açıklanmıştır. Dördüncü bölümde, Python programı kullanılarak yapılan uygulama sonucu elde edilen bulgular tartışılmıştır. Son olarak sonuç ve öneriler kısmında COVID-19 analizi ile elde edilen sonuçlar değerlendirilerek gelecek çalışmalar için önerilerde bulunulmuştur.

2. LİTERATÜR ARAŞTIRMASI

Geçmişteki birçok salgın hastalıkta olduğu gibi bugünde araştırmacılar COVID-19 salgınının yayılımını modellemek ve epidemiyolojik eğrileri tahmin etmek için çeşitli çalışmalar yapmaktadır [10]. Bu çalışmalar kapsamında COVID-19 salgınının gelecekteki yayılma hızını öngören uyarı sistemleri [3], yetkililerin salgının yayılımı sonucunda oluşacak durumlar ile mücadelede stratejiler belirlemesi amacıyla çeşitli uygulamaların geliştirilmesi ve COVID-19'un eğitim, ulaşım, ekonomi gibi farklı sektörler üzerindeki etkisinin tahmini gibi konular ele alınmıştır [11].

Genel olarak COVID-19 için literatürde tahmine dayalı modellemelerde duyarlı-enfekte-kurtarılmış (Susceptible-Infected-Recovered) (SIR), duyarlı-maruz kalmış-enfekte-kurtarılmış (Susceptible-Exposed-Infected-Recovered) (SEIR) gibi etmen tabanlı modeller ve eğri uydurma modelleri kullanır. Salgın tahmin modellerinde hangi faktörlerin dikkate alındığına bağlı olarak, tahminler kısa vadeli ve uzun vadeli olabilir. Santosh 2020 yılındaki çalışmasında, rastgele oluşturulmuş parametreler ile stokastik modeller, ayrık modeller, buna benzer sürekli ve sıra dışı faktörlerin karmaşık tahmin modelleri tasarlamayı sağladığını göstermiştir [12].

COVID-19 için çeşitli salgın tahmin modelleri ise, dünyanın dört bir yanındaki yetkililer tarafından bilinçli kararlar almak ve ilgili kontrol önlemlerini uygulamak için kullanılmaktadır. Bu amaçla küresel COVID-19 yaygınlığını tahmin etmek için standart modeller arasında epidemiyolojik ve istatistiksel modeller araştırmacılar tarafından genel olarak tercih edilse de istatistiksel teknikler üzerine kurulu makine öğrenimi modelleri daha fazla ilgi görmüştür. Makine öğrenimi modelleri yüksek düzeyde belirsizlik ve veri eksikliği nedeniyle, standart modeller göre uzun vadeli tahminler için yüksek tahmin doğruluğu göstermişlerdir [13].

Jubin ve diğerleri 2021 yılında yaptıkları bir çalışmada, literatürde yer alan SIR ve SEIR epidemiyolojik salgın modellerine alternatif olarak Hindistan’da ve dünyada COVID-19 yaygınlığını tahmin etmek amacıyla makine öğrenimi tekniklerinden lineer regresyon ve destek vektör makinesi ile karşılaştırmalı bir çalışma gerçekleştirmişlerdir. Elde edilen sonuçlara göre lineer regresyon modelinin salgını tahmin etmede daha etkili olduğu görülmüştür [14].

Uluslararası birçok çalışmanın yanı sıra Türkiye için de uygulanan COVID-19 tahmin modelleri mevcuttur. Bu çalışmaları genel olarak salgının matematiksel modeller ve makine öğrenimi modelleri olarak gruplandırabiliriz.

Matematiksel modelleme çalışması olarak birleştirilmiş sıradan diferansiyel denklem (ODE'ler) sistemi kullanarak COVID-19 salgınının temel üreme sayısı hesaplanarak analiz edilmiştir. Aslan ve ark. 2020 yılında, Çin'in Hubei eyaletini bir test ortamı olarak kullanarak, COVID-19 vakaları düşmeye başladıktan sonra geriye kalan büyük miktarda veriyi COVID-19'un temel üreme sayısı olan R_0'ı, Hubei salgınının toplam vakalarını, toplam ölümlerini tahmin etmişlerdir. Sayısal deneyler yoluyla karantina, sosyal mesafe ve COVID-19 testinin salgının dinamikleri üzerindeki etkilerini gözlemlemişlerdir. Hubei salgınından toplanan bilgileri kullanarak, Türkiye'deki salgının dinamiklerini analiz etmek için bir matematiksel model geliştirmişlerdir. Farklı seviyelerde sosyal mesafe, karantina önemleri için salgının zirvesi ve Türkiye'deki toplam vaka/ölüm sayısı için tahminler gerçekleştirmişlerdir [15].

Dünya geneli yaygın olarak tercih edilen SIR epidemiyolojik tahmin modelini Özdinç ve ark. 2020 yılında, Türkiye için COVID-19 salgınının başladığı günden sonraki 43 günlük vaka sayısı ile uygulamıştır. Temel üreme sayısı (R_e)' ı 2,4 olarak tahmin etmiş ve aktif olarak enfekte olmuş maksimum vaka sayısının zirve tarihine ilişkin model çıktılarını sunmuşlardır. Salgının ilerleyen dönemlerinde model çıktılarının karşılaştırılmasında sosyal mesafe önlemlerinin olumlu etkisini belirtmişler ve salgının Türkiye'de zirve noktası ile ilgili tahmin sonuçlarını değerlendirmişlerdir [16].

2020 yılında Arslan ve diğerleri salgın tahmin çalışmalarını üç aşamada gerçekleştirmişlerdir. İlkinde, COVID-19 vaka sayısını tahmin etmişlerdir. İkincisinde, müdahale yapılmaması durumunda yani karantina önlemleri olmadan beklenen toplam enfekte kişi sayısı, ölümler, hastaneye yatışlar tahmin edilmiştir. Son bölümde, beklenen enfekte kişi sayısı, ölüm sayısı, yoğun bakım ve yoğun bakım dışı yatak ihtiyaçlarının zaman içindeki dağılımı SEIR tabanlı bir simülatör olan TÜRKSAŞ aracılığıyla farklı senaryolarla tahmin edilmiştir [17].

Literatürde yer alan bazı çalışmalar ise, yapay zekâ tabanlı makine öğrenimi algoritmalarının COVID-19 yayılım modellerine uygulanmasının küresel bilim topluluğunun daha fazla

ilgisini çektiğini göstermektedir. Makine öğrenimi yöntemleri ile COVID-19'un yayılım modellerini kapsamlı olarak incelemek için çeşitli veri tabanlarından bir elektronik literatür taraması yapılmıştır. Gözden geçirilmiş makalelerden yola çıkarak makine öğrenimi tabanlı tahmin algoritmalarının COVID-19 salgınının yayılım modellemesinde açık bir uygulamaya sahip olduğu ve pandemilere karşı mücadelede önemli bir araç olabileceği sonucuna varılmıştır. Bu araçlardan sıkça tercih edilen tahmin yöntemlerden birisi de regresyon analizidir [18].

COVID-19 verileri ile yapılan tahmin çalışmalarının bir kısmı sadece regresyon yöntemleri uygulamalarıdır [19-21]. Örneğin regresyon temelli bir makine öğrenimi modelinde farklı ülkelerin verilerinden yararlanılarak COVID-19 vakalarının büyüklüğünü tahmin etmeye yönelik bir çalışma sunulmuştur [22]. Regresyon teknikleri ile oluşturulan yayılım modellerinin diğer bir kısmı ise tahmin modellerinin karşılaştırılması biçiminde oluşan çalışmalardır [23-24].

Yeşilkanat ve diğerleri (2020), RFR makine öğrenme algoritmasının dünyadaki 190 ülke için yakın gelecekteki COVID-19 vaka sayılarının tahmin edilmesindeki performansı araştırılmış ve gerçek teyit edilmiş vaka sonuçlarıyla karşılaştırmalı olarak haritalanmıştır. 23/01/2020-17/06/2020 tarihleri arasında teyit edilen vakaları; eğitim alt verisi, test alt verisi ve tahmin alt verisi olarak 3 ana alt veri kümesine bölerek RFR modelini oluşturmuşlardır. Çalışma sonunda RFR modeli tahminlerinin alt verilerini test etmek için determinasyon katsayısı (R^2) değerlerinin 0,843 ile 0,995 arasında (ortalama R^2= 0,959) ve ortalama hata kareleri karekökü (RMSE) değerlerinin 141,76 ile 526,18 arasında (ortalama RMSE = 259,38) olduğu ve alt verileri tahmin etmek için R^2 değerlerinin 0,690 ile 0,968 arasında (ortalama R^2 = 0,914) ve RMSE değerlerinin 549,73 ile 2500,79 arasında (ortalama RMSE = 909,37) olduğu bulunmuştur. Bu sonuçlar, COVID-19 gibi aniden ortaya çıkan ve hızla yayılan bir salgın durumunda, RFR makine öğrenme algoritmasının yakın gelecek için vaka sayısını tahmin etmede iyi performans gösterdiğini göstermektedir [25].

Prakash ve diğerleri (2020), COVID-19'dan en çok hangi yaş grubunun etkilendiğini anlamak için COVID-19 veri setleri üzerinde bir analiz yapmıştır. Makine öğrenimi algoritmaları kullanılarak farklı tahmin modelleri oluşturulmuş ve performansları hesaplanmıştır. RFR modeli R^2=0,97 değeri ile rastgele orman sınıflandırıcısı, destek vektör makinesi, karar ağacı sınıflandırıcısı, Gauss naive bayes sınıflandırıcı, çoklu doğrusal

regresyon, lojistik regresyon ve eXtreme Gradyan Artırma (XGBoost) sınıflandırıcısı gibi makine öğrenimi modellerinden daha iyi performans göstermiştir [26].

Gupta ve diğerleri (2021), COVID-19'un Hindistan eyaletleri ve birlik toprakları üzerindeki etkisinin kapsamlı analizine ve her eyalette taburcu edilen hasta ve ölüm sayısını tahmin etmek için regresyon modelleri geliştirilmiştir. Önerilen modellerin performansı polinom regresyonu, karar ağacı regresyonu ve RFR kullanılarak belirlenmiştir. Sonuçlar, RMSE değeri % 0,08 olan polinom regresyonunun taburcu edilen hastaların tahmininde, % 0,14 RMSE değeri ile RFR'nin ise ölümlerin tahmin edilmesinde diğer yöntemlere kıyasla daha etkili bir performans göstermiştir [27].

Muhammad ve diğerleri (2020), Güney Kore'deki COVID-19 hastalarının epidemiyolojik veri seti kullanılarak COVID-19 ile enfekte hastaların iyileşme durumlarının tahmini için modeller geliştirmişlerdir. Karar ağacı, destek vektör regresyon, naive bayes, lojistik regresyon, RFR ve k-en yakın komşu algoritmaları ile modeller geliştirerek tahmin performanslarını karşılaştırmışlardır [28].

Pan ve diğerleri (2021), COVID-19'un neden olduğu küresel salgınla mücadelede yetkilileri bilgilendirmek için bir topluluk modeli olan RFR ile çok amaçlı bir optimizasyon algoritmasının (NSGA-II) kombinasyonunu yaparak bir uzay-zamansal analiz çerçevesi geliştirmişlerdir. Model; Japonya, Güney Kore, Pakistan ve Nepal dahil olmak üzere dört Asya ülkesi için doğrulanmıştır. Uygun bir zaman penceresine sahip RFR, onaylanmış vakaların günlük büyümesini ve ulusal ölçekte günlük ölüm oranını doğru bir şekilde tahmin etmek için çevresel ve sosyal faktörlerin birleşik etkilerini öğrenebilir ve bunu bir dizi Pareto bulmak için NSGA-II takip ederek uygun çözümler sunmuştur [29].

Vaishnav ve diğerleri (2021), COVID-19 sırasında bildirilen toplam vaka, toplam iyileşme ve toplam ölüm açısından COVID-19'un etkisini tahmin etmek amacıyla bir analitik analiz yapmışlardır. Bu analiz için Hindistan eyaletleri seçilmiştir ve tüm araştırma çalışmaları dünya sağlık örgütü resmi platformunda elde edilen veri setleri analiz edilmiştir. Bu analizi gerçekleştirmek için makine öğrenimi algoritmalarından RFR ve karar ağacı regresyon modelini kullanmışlardır [30].

COVID-19 vakalarının büyüklüğünü tahmin etmek için bazı tahmin yöntemlerine ihtiyaç duyulmaktadır ve şimdiye kadar farklı tahmin yöntemlerine ilişkin çok sayıda çalışma sunulmuştur. Saqib 2020 yılındaki çalışmasında, yalnızca iyi bir doğrulukla tahmin edilen değil, aynı zamanda tahminlerin belirsizliğini de dikkate alan bir hibrit makine öğrenme modeli önermiştir. Model, n-dereceli bir polinom ile hibritleştirilmiş BRR kullanılarak formüle edilmiştir ve geleneksel yöntemler yerine bağımlı değişkenin değerini tahmin etmek için olasılık dağılımını kullanmıştır. Sonuçları doğrulamak için Amerika Birleşik Devletleri, İtalya ve İspanya örnek vakalarını tahmin etmiştir. Hibrit PBRR modeli test verilerinde %91 doğrulukla dünya çapındaki verilerde 5. Derece polinom ile etkili bir tahminde bulunmuştur [22].

Malki ve diğerleri (2020), farklı faktörler ve COVID-19'un yayılma hızı arasındaki ilişkiyi ortaya çıkarmak için çeşitli makine öğrenimi modelleri önermişlerdir. Bu çalışmada kullanılan makine öğrenimi algoritmaları, teyit edilen vaka sayısı ile belirli bölgelerdeki hava durumu değişkenleri arasındaki ilişkiyi çıkararak, sıcaklık ve nem gibi hava değişkenlerinin COVID-19'un bulaşması üzerindeki etkisini tahmin edilmiştir. RFR; R^2: 0,68 değeri ile BRR; R^2: 0,72 değer yöntemlerden daha iyi olarak bulunmuştur [31].

Parhusip (2020), yakın gelecekte başta Endonezya olmak üzere COVID-19 vakalarının sayısını tahmin etmiştir. Çalışmada destek vektör makinesi ve BRR ile model kurmuştur. Çeşitli ülkelerin vaka sayısı ile gerçekleştirdiği çalışmada 10 gün ilerisini daha iyi tahmin eden BRR olmuştur [32].

Ali (2020), COVID-19 hastalığının bulaşma ölçeğini, iyileşme oranını ve ölüm oranını tahmin etmiştir. Destek vektör makinesi, BRR, polinom regresyon ve SIR epidemiyolojik modeli gibi matematiksel modelleme tekniklerinin yanında makine öğrenimi modellerini de uygulamıştır. Destek vektör makinesi R^2 puanı = 0,8273, polinom regresyon R^2 = 0,90769 ve BRR R^2 = 0,93213 değerlerine ulaşmıştır. BRR üç model arasında en iyi performansı göstermiştir [33].

Khakharia ve diğerleri (2021), yüksek ve yoğun nüfuslu ilk 10 ülke için COVID-19 için bir salgın tahmin sistemi geliştirmişlerdir. Önerilen tahmin modelleri, 9 farklı makine öğrenimi algoritması kullanarak art arda 5 gün boyunca ortaya çıkması muhtemel yeni vakaların

sayısını tahmin etmiştir. Çalışmada BRR modeli doğruluk skoru %92,94, RFR için doğruluk skoru %88,44 olarak bulunmuştur [34].

2020 yılındaki bir çalışmada ise Taşdelen ve diğerleri, Türkiye'de COVID-19 salgını nedeniyle uygulanan karantina önlemlerinin, test sayısı ve vaka sayısı üzerine etkisini poisson regresyon tekniği ile analiz etmişlerdir [35].

COVID-19 yaygınlığının tahmini ile ilgili literatürdeki çalışmalara bakıldığında regresyon analizi konusunda RFR ve BRR tekniklerinin yüksek tahmin performansları ile ön plana çıktığı görülmektedir. Ayrıca COVID-19 yaygınlığının tahmin edilmesinin ötesinde farklı sektörlerde çeşitli tahmin problemleri ile de RFR ve BRR makine öğrenimi algoritmalarının etkili uygulamaları söz konusudur. RFR ve BRR yöntemleri ile ilgili literatürde yer alan çalışmalar yayınlandıkları yıllar göz önünde bulundurularak aşağıda özetlenmiştir.

Adusumilli ve diğerleri (2015), GPS sinyal kesintilerini önlemek amacıyla Küresel Konumlandırma Sistemi (GPS) ve Ataletsel Navigasyon Sistemi (INS) verilerini entegre etmek için ilk kez hibrit bir Temel Bileşen Regresyonu (PCR) ve RFR yöntemini önermişlerdir. PCR, verilerde mevcut olan doğrusallığı modellerken RFR, PCR'nin yakalayamadığı geriye kalan doğrusal olmayan kısmı modeller. Bu hibrit yaklaşımın hem doğrusal regresyon hem de doğrusal olmayan regresyon modellerinin avantajlarını bir araya getirilerek, mevcut RFR yöntemine kıyasla RMSE tarafından verilen konumsal sapma ve konumlama hatasında önemli bir azalmaya yol açtığı gözlemlenmiştir. Farklı koşullar ve değişen süreler altında farklı kesintiler için önerilen model, RFR modeline kıyasla tahmin doğruluğunda toplam %14–%45 iyileşme göstermiştir [36].

Wirkert ve diğerleri (2016), modern yüksek çözünürlüklü laparoskopik görüntülere uygun fizyolojik parametreleri doğru ve hızlı bir şekilde tahmin etmek için bir yöntem geliştirmekti. Monte Carlo simülasyonları ile oluşturulan yansıma spektrumlarını kullanarak RFR eğitilmiştir. Deneylere göre, yöntem en son teknoloji ürünü çevrimiçi yöntemlerden daha yüksek doğruluk ve sağlamlığa sahiptir aynı zamanda diğer doğrusal olmayan regresyon tabanlı yöntemlerden çok daha hızlıdır [37].

Ahmad ve diğerleri (2018), bir güneş enerjisi termal kollektöründen gelen faydalı saatlik enerjiyi tahmin etmek için makine öğrenimi modellerinden RFR, karar ağaçları ve destek

vektör regresyonunu karşılaştırarak kapsamlı bir çalışma sunmuştur. Geliştirilen modeller; genelleme yetenekleri, doğruluk ve hesaplama maliyetlerine göre karşılaştırılmıştır. RFR ve ET (extra trees regressor)'nin karşılaştırılabilir tahmin gücüne sahip olduğu ve faydalı güneş termal enerjisini tahmin etmek için eşit derecede uygulanabilirdir ve RMSE değerleri sırasıyla 6,86 ve 7,12 ile diğerlerinden daha düşüktür [38].

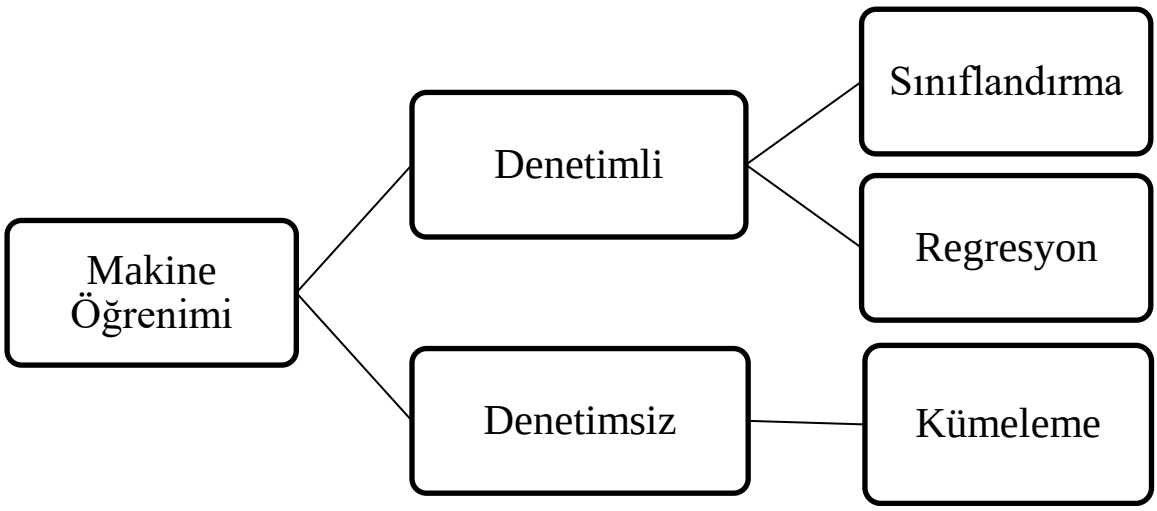
Singh ve diğerleri (2018), RFR kullanılarak toprağın sızma hızı tahmin edilmiş ve performansları YSA ve M5P model ağaç teknikleri ile karşılaştırılmıştır. 132 alan ölçümünden oluşan bir veri seti kullanılmış ve rastgele seçilen 132 gözlemden 88'i eğitim için, geri kalan 44'ü ise modeli test etmek için kullanılmıştır. Sonuçlar, RFR yaklaşımının diğer YSA ve M5P model ağacından daha iyi çalıştığını göstermektedir [39].

Jakariya ve diğerleri (2020), tarımsal kırılganlığı ölçmek için Geçim Kırılganlığı Endeksi'ni (LVI) kullanarak iklimsel faktörler dışındaki sorunları belirlemeye odaklanmışlardır. Tarımsal güvenlik açığını değerlendirerek en önemli faktörleri tespit etmek için iklime dayanıklı bir geçim kaynağı güvenlik açığı değerlendirme aracının geliştirilmesini makine öğrenimi tekniklerini kullanarak yapmışlardır. Makine öğrenimi yöntemlerini kullanarak, daha az değişkenle gerçeğe daha yakın güvenlik açığı puanları oluşturacak daha iyi bir yaklaşım bulmayı seçmişlerdir. R^2 değerlerini BRR: 0,91 ve RFR: 0,83 olarak bulmuşlardır [40].

Bu çalışmada ise Türkiye'de COVID-19 yaygınlığının dinamikleri incelenmiştir. İstatistiki bölgelerin günlük COVID-19 vaka sayısı verileri kullanılarak RFR, BRR ve LR yöntemleri ile tahmin modelleri oluşturulmuştur. Çalışmada literatürde tahminleme ile ilgili yer alan uygulamalarda etkili performans gösteren rastgele orman regresyonu (RFR), bayes sırt regresyon (BRR) ve lineer regresyon (LR) algoritmaları Türkiye'de COVID-19 salgınının yaygınlığını tahmin etmek amacıyla kullanılmıştır. Bu çalışmanın literatüre katkısı, Türkiye'nin 12 istatistiki bölgesi için COVID-19 vaka sayısını tahmin etmek üzerine modellerin ayrı ayrı kurulmasıdır. Bölge bazlı vaka sayılarının tahmin edilmesindeki esas amaç farklı iklim ve farklı demografi koşullarının salgınların genel yayılımını etkileyen ana unsurlardan olmasıdır. Ülkeler geneli tahmin modellerinde bölgesel yayılım farklılıklarının oluşturabileceği kısıtlar göz ardı edildiğinden, bölge bazlı tahminlerde daha gerçekçi ve net tahmin modelleri söz konusu olmasıdır.

3. MAKİNE ÖĞRENİMİ YÖNTEMLERİ

Verilerden öğrenme, makine öğrenimi modelleri ile klasik algoritmalar arasındaki temel farktır. Klasik algoritmalar, karmaşık bir sistemde en iyi cevabın nasıl bulunacağını ifade eder ve algoritma daha sonra bu en iyi çözümleri arar, genellikle bir insandan daha hızlı ve daha etkili çalışırlar. Makine öğrenimi ise, bir korelasyonlar yani ilişkiler oyunudur. Mevcut olan çoğu makine öğrenimi algoritması, veri setleri arasındaki ilişkileri bulmak veya bulunan bu ilişkilerden yararlanmakla ilgilenir. Makine öğrenimi algoritmaları belirli korelasyonları tespit edebildiğinde, model bu ilişkileri gelecekteki gözlemleri tahmin etmek için kullanabilir veya ilginç kalıpları ortaya çıkarmak için verileri genelleştirebilir. Makine öğrenimi algoritmaları, verileri analiz edebildikleri ve veri setleri içerisindeki örüntüleri tanıyabildikleri kadarıyla makineleri kendi kendine öğrenebilir hale getirmeyi amaçlarlar. Daha az varsayım ve insan çabası gerektirse de güçlü bir tahmin yeteneğine sahiptirler [41].



Şekil 3.1. Makine öğrenimi türleri

Şekil 3.1'de görüldüğü gibi makine öğrenimi yöntemleri genel olarak denetimli ve denetimsiz öğrenme olarak sınıflandırılır [42]. Denetimli makine öğrenimi algoritması, bir girdi değişken kümesini ve değişkenlere bilinen cevapları alır, arkasından yeni değişkenlere cevap için gerçeğe yakın tahminler elde etmek amacıyla modeli eğitir. Denetimli makine öğrenimi modelleri genellikle tahmine dayalı analitik modeller olarak adlandırılır ve bu modeller, geçmişe dayalı olarak geleceği tahmin eder. Bilinen bir eğitim veri setini araştırarak, geçmiş verilere dayalı olarak çıktı değerleri hakkında bir fikir sahibi olmak için

ortalama bir işlev üretir. Yanıt bazen çıktı, etiket, hedef ve bağımlı değişken olarak adlandırılır [41].

Müşteri kaybının azaltılması, müşteri yaşam boyu değerinin belirlenmesi, ürün önerilerinin kişiselleştirilmesi, insan kaynaklarının dağıtılması, satışların öngörülmesi, arz ve talebin tahmin edilmesi, dolandırıcılığın belirlenmesi ve ekipman bakımlarının öngörülmesi denetimli öğrenmenin uygulama alanlarıdır [20].

Denetimsiz öğrenme, denetimli öğrenmeye göre daha karmaşık algoritmalara sahiptir. Denetimsiz öğrenmenin büyük bir avantajı, etiketlenmiş veri gerektirmemesidir. Bu nedenle denetimsiz öğrenme modellerinde uygun verilerin elde edilmesi daha kolaydır. Denetimli öğrenme etiketli verilerle çalışırken denetimsiz öğrenme etiketlenmemiş verilerle çalışır [43].

3.1. Kümeleme

Kümeleme algoritması, verileri anlamlı, faydalı veya her ikisi birden olan gruplara (kümelere) böler. Hedef anlamlı gruplar oluşturmaksa, kümeler verilerin doğal yapısını yakalamalıdır. Ancak bazı durumlarda, verilerin boyutunu küçültmek için veriyi özetleme yoluyla kümeleme yöntemi kullanılır. Kümeleme yöntemi uzun zamandan beri birçok sektörde uygulama alanına sahiptir. Psikoloji ve diğer sosyal bilimler, biyoloji, istatistik, örüntü tanıma, makine öğrenimi gibi alanlarda sıkça tercih edilen yaklaşımlardan biridir.

Kümeleme analizi, veri nesnelerini, yalnızca nesneleri ve bunların ilişkilerini tanımlayan verilerde bulunan bilgilere dayalı olarak gruplandırır. Amaç, bir küme içerisindeki nesnelerin birbirine benzer (veya ilişkili) ve diğer gruplardaki nesnelerden farklı (veya ilgisiz) olmasıdır. Bir küme içerisindeki homojenlik ne kadar büyükse ve kümeler arasındaki farklılık ne kadar fazlaysa, kümeleme o kadar iyi ve belirgindir [44].

3.2. Sınıflandırma

Sınıflandırma, etiketlenmemiş veri örneklerine etiket atama görevidir ve böyle bir görevi gerçekleştirmek için bir sınıflandırıcı kullanılır. Bir sınıflandırıcı tipik olarak önceki bölümde gösterildiği gibi bir model açısından tanımlanır. Model, her bir örnek için sınıf

etiketlerinin yanı sıra ceza değerleri içeren eğitim kümesi olarak bilinen belirli bir örnek kümesi kullanılarak oluşturulur. Bir eğitim seti verilen bir sınıflandırma modelini öğrenmeye yönelik sistematik yaklaşım, öğrenme algoritması olarak bilinir. Eğitim verilerinden bir sınıflandırma modeli oluşturmak için bir öğrenme algoritması kullanma süreci, tümevarım olarak bilinir. Bu süreç genellikle "bir model öğrenmek" veya "bir model oluşturmak" olarak da tanımlanır. Sınıf etiketlerini tahmin etmek için görünmeyen test örneklerine bir sınıflandırma modeli uygulama süreci, tümdengelim olarak bilinir. Bu nedenle, sınıflandırma süreci iki adımı içerir: bir modeli öğrenmek için eğitim verilerine bir öğrenme algoritması uygulamak ve ardından etiketlenmemiş örneklere etiket atamak için modeli uygulamak [45].

3.3. Regresyon

Regresyon, iki teori için kullanılan bir tekniktir. İlk olarak, regresyon analizleri genellikle, uygulamalarının makine öğrenimi alanıyla büyük örtüşmelere sahip olması nedeniyle tahmin için kullanılır. İkinci olarak, bazı durumlarda bağımlı ve bağımsız değişkenler arasındaki nedensel ilişkileri belirlemek için regresyon analizleri kullanılabilir. Bunlardan daha önemlisi, tek başına regresyonlar yalnızca bağımlı bir değişken ile farklı değişkenlerden oluşan sabit bir veri kümesi koleksiyonu arasındaki ilişkileri gösterir. Regresyon modellerine göre, bağımsız değişkenler bağımlı değişkenleri öngörmektedir [46].

Regresyon analizi, pratikte en sık kullanılan istatistiksel yöntemlerden biridir. Regresyon analizi uygulamaları tıp, biyoloji, tarım, ekonomi, mühendislik, sosyoloji, jeoloji vb. dahil olmak üzere birçok bilimsel alanda bulunabilir.

Regresyon analizinin amaçları aşağıda listelenmiştir:

1. Hedef değişken y ile $x_1, x_2, \cdots, x_n$ arasında geçici bir ilişki kurmak,
2. $x_1, x_2, \cdots, x_n$ değerlerine dayalı olarak y'yi tahmin etmek,
3. Nedensel ilişkinin daha verimli ve doğru bir şekilde belirlenebilmesi için y yanıt değişkenini açıklamak amacıyla hangi değişkenlerin diğerlerinden daha önemli olduğunu belirleyerek $x_1, x_2, \cdots, x_n$ değişkenlerini bulmaktır.

3.3.1. Lineer regresyon

Lineer regresyon (LR), modelin regresyon parametrelerinin doğrusal olmasını gerektirir. Regresyon analizi, bir veya daha fazla hedef değişkeni (bağımlı değişkenler, açıklanan değişkenler, tahmin edilen değişkenler, genellikle y ile gösterilir) ile tahmin ediciler (bağımsız değişkenler, açıklayıcı değişkenler, kontrol değişkenleri, genellikle x_1, x_2, $\cdots$, x_p ile gösterilir) arasındaki ilişkiyi keşfetme yöntemidir.

Lineer regresyonun üç türü vardır. Birincisi basit lineer regresyondur. Basit lineer regresyon, iki değişken arasında doğrusal bir ilişki olduğunu varsayar. Bu değişkenlerden biri bağımlı değişken y, diğeri ise bağımsız değişken x'tir. Mesela, basit lineer regresyon, kas gücü (y) ile yağsız vücut kütlesi (x) arasındaki ilişkiyi modelleyebilir. Basit lineer regresyon modeli genellikle aşağıdaki biçimde yazılır:

$$y = \beta_0 + \beta_1 x + \varepsilon \tag{3.1}$$

Eş. 3.1'de görüldüğü gibi y bağımlı değişken, β_0; x=0 olduğunda grafiğin y'yi kestiği nokta, β_1 regresyon çizgisinin eğimi, x bağımsız değişkendir ve ε rastgele hatadır. Basit bir doğrusal regresyonda genellikle ε hatasının $E(\varepsilon) = 0$ ve sabit varyans $Var(\varepsilon) = \sigma^2$ ile normal olarak dağıldığı varsayılır.

İkinci tip doğrusal regresyon, tek bağımlı değişkenli ve birden fazla bağımsız değişkenli doğrusal regresyon modeli olan çoklu doğrusal regresyondur. Çoklu doğrusal regresyon, hedef değişkenin model parametrelerinin doğrusal bir fonksiyonu olduğunu ve modelde birden fazla bağımsız değişken olduğunu varsayar. Çoklu doğrusal regresyon modelinin genel formu aşağıdaki gibidir:

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p + \varepsilon \tag{3.2}$$

Eş. 3.2'de yer alan y bağımlı değişken, β_0, β_1, β_2, $\cdots$, β_p regresyon katsayılarıdır ve x_1, x_2, $x_3 \cdots$, x_n modeldeki bağımsız değişkenlerdir. Klasik regresyon modelinde genellikle hata teriminin ε'nin $E(\varepsilon) = 0$ ve sabit bir varyans $Var(\varepsilon) = \sigma^2$ ile normal dağılımı takip ettiği varsayılır.

Basit lineer regresyon, bir bağımlı değişken ile bir bağımsız değişken arasındaki doğrusal ilişkiyi araştırırken, çoklu doğrusal regresyon, bir bağımlı değişken ile birden fazla bağımsız değişken arasındaki doğrusal ilişkiye odaklanır. Çoklu doğrusal regresyon, ortak doğrusallık, varyans şişirme, regresyon teşhisinin grafiksel gösterimi ve regresyon aykırı değerinin tespiti ve etkili gözlem gibi basit doğrusal regresyondan daha fazla konu içerir.

Üçüncü tip regresyon, bağımlı değişken ile bağımsız değişkenler arasındaki ilişkinin regresyon parametrelerinde lineer olmadığını varsayan lineer olmayan regresyondur. Doğrusal olmayan regresyon modeli (büyüme modeli) örneği şu şekilde yazılabilir:

$$y = \frac{\alpha}{(1 + e^{\beta t})} + \varepsilon \tag{3.3}$$

Eş. 3.3'te ise y, belirli bir organizmanın t zamanının bir fonksiyonu olarak büyümesidir, α ve β model parametreleridir ve ε rastgele hatadır. Doğrusal olmayan regresyon modeli, model parametrelerinin tahmini, model seçimi, model teşhisi, değişken seçimi, aykırı değer tespiti veya etkili gözlem tanımlaması açısından doğrusal regresyon modelinden daha karmaşıktır [47].

3.3.2. Lojistik regresyon

Araştırmaların çoğunda olaylar arsındaki nedensellik ilişkilerinin belirlenebilmesi amacıyla ele alınan değişkenler olumlu-olumsuz, başarılı-başarısız, evli-bekar, memnun-memnun değil şeklindeki iki düzeyli yapılardan oluşmaktadır. Bağımlı değişkenlerin iki düzeyli veya çok düzeyli kategorik verilerden oluşması halinde; bağımlı değişken ile bağımsız değişken arasındaki neden-sonuç ilişkisinin belirlenmesinde lojistik regresyon yöntemi önemli bir yere sahiptir.

Lojistik regresyon, sınıf koşullu olasılıkları hakkında herhangi bir varsayımda bulunmadan poster olasılıklarını doğrudan hesaplayan sınıflandırma için ayırt edici bir modeldir. Bu nedenle, oldukça geneldir ve çeşitli uygulamalarda kullanılabilir. Ayrıca, çok terimli lojistik regresyon olarak bilinen çok sınıflı sınıflandırmaya kolayca genişletilebilir. Bununla birlikte, ifade gücü yalnızca doğrusal karar sınırlarını öğrenmekle sınırlıdır.

Her öznitelik için farklı ağırlıklar (parametreler) olduğundan, öznitelikler ve sınıf etiketleri arasındaki ilişkileri anlamak için öğrenilen lojistik regresyon parametreleri analiz edilebilir.

Lojistik regresyon, öznitelik uzayındaki hesaplama yoğunluklarını ve mesafeleri içermediği için, en yakın komşu sınıflandırıcıları gibi uzaklık tabanlı yöntemlere göre yüksek boyutlu ayarlarda bile daha sağlam çalışabilir. Bununla birlikte, lojistik regresyonun amaç fonksiyonu, modelin karmaşıklığına ilişkin herhangi bir terim içermez. Bu nedenle, lojistik regresyon, destek vektör makineleri gibi diğer sınıflandırma modelleriyle karşılaştırıldığında, model karmaşıklığı ve eğitim performansı arasında bir denge kurmanın bir yolunu sağlamaz. Bununla birlikte, çapraz entropi fonksiyonu ile birlikte amaç fonksiyonuna uygun terimler dahil edilerek, model karmaşıklığını hesaba katmak için lojistik regresyon çeşitlerine kolaylıkla dönüştürülebilir [1].

Lojistik regresyon, sınıflandırma ve atama problemlerini çözmeye yardımcı olan bir regresyon tekniğidir. Normal dağılım ve verilerin sürekliliği varsayımı koşulu söz konusu değildir. Bağımlı değişken üzerinde açıklayıcı değişkenlerin etkileri olasılık olarak elde edilerek, risk faktörlerinin olasılıksal olarak belirlenmesi sağlanır ve aşağıdaki eşitlik ile gösterilir [48]:

$$P = \frac{e^{\beta_0 + \beta_1 x_1 + \ldots\ldots \beta_i x_i}}{1 + e^{\beta_0 + \beta_1 x_1 + \ldots\ldots \beta_i x_i}} \tag{3.4}$$

Eş. 3.4'te yer alan notasyonlar:

P: İncelenen olayın gözlenme olasılığını,
β_0: Bağımsız değişkenler sıfır değerini aldığında bağımlı değişkenin değerini başka bir ifadeyle sabiti,
β_1 , β_2….. β_k: Bağımsız değişkenlerin regresyon katsayılarını,
x_1, x_2…. x_k: Bağımsız değişkenleri,
k: Bağımsız değişken sayısını,
e: 2.71 sayısını göstermektedir.

3.3.3. Polinom regresyon

Polinom regresyon, yalnızca bir bağımsız değişken x ile çoklu regresyonun özel bir durumudur. Bir tahmin edici ile hedef değişken arasındaki ilişki doğrusal olmadığında polinom regresyon tercih edilir. Tek değişkenli polinom regresyon modeli şu şekilde ifade edilebilir [49]:

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \cdots + \beta_k x_i^k + \varepsilon_i \qquad i=1,2,\ldots\ldots\ldots n \tag{3.5}$$

Eş. 3.5'te k, polinomun derecesidir. Polinom derecesi arttıkça, model doğrusal olmayan ilişkilere daha iyi uyum sağlar, ancak pratikte k'nin 3 veya 4'ten büyük olmasını nadir tercih edilen bir durumdur. Bu noktanın ötesinde, model çok esnek hale gelir ve verilere fazla uyar.

3.3.4. Sırt (Ridge) regresyon

Düzenlileştirme, makine öğrenimi modellerinde aşırı uydurma (overfitting) sorununun üstesinden gelmeye yardımcı olan bir tekniktir. Parametreleri düzenli veya normal tutmaya yardımcı olduğu için düzenlileştirme denir. Yaygın teknikler genel olarak lasso ve sırt (ridge) regresyonu olarak bilinen sırasıyla L1 ve L2 düzenlileştirmeleridir.

Sırt regresyonu, çoklu doğrusal regresyonda sıklıkla ortaya çıkan ortak doğrusallık problemini ele almak için kullanılan popüler bir parametre tahmin yöntemidir. Sırt metodolojisinin formülasyonu gözden geçirilir ve sırt tahminlerinin özellikleri birleştirilir. Özellikle sırt formunun bir regresyon tahmin edicisine yol açan dört gerekçe özetlenmiştir. sırt parametresinin küçük değerleri (en küçük kareler çözümüne yakın olan değerler) ve sırt parametresinin büyük değerleri için bir sırt izinin davranışını açıklayan Sırt regresyon katsayılarının cebirsel özellikleri verilmiştir [50].

Sırt parametresinin bir fonksiyonu tahmin edilen regresyon katsayılarının toplamıdır. Aslında sırt regresyon metodolojisi, negatif olmayan bir sayısal (skaler) parametre tarafından indekslenen bir yanlı tahmin ediciler sınıfı verir. Buradaki zorluk, belirli bir problem bağlamında bu sınıf içinde hangi tahmin parametresinin kullanılacağını belirlemek, yani sırt parametresi için en iyi seçimi belirlemektir. Sırt regresyon eşitliği [51]:

$$\hat{\beta} = (x'x)^{-1}x'y \tag{3.6}$$

Çoklu bağlantı söz konusu olduğu zaman bağımsız değişkenler arasındaki korelasyonun yüksek çıkması $(x'x)$ matrisinin varyanslarını ciddi derecede arttıracaktır. Bunun sonucu olarak da gerçekte önemli olan parametre değerleri varyansın artması nedeniyle önemsiz gözükecektir. Bu sorunu gidermek amacıyla Eş. 3.6'da yer alan $(x'x)$ matrisinin köşegen elemanlarına pozitif bir k sabiti eklenerek bu matrisin varyansı düşürülür [52-53].

Sırt regresyonu için parametre tahmini Eş. 3.6'ya k sabiti eklenirse aşağıdaki eşitlik elde edilir:

$$\hat{\beta} = (x'x + kı)^{-1}x'y \qquad 0 \leq k \leq 1 \tag{3.7}$$

Eş. 3.7'de k değeri sıfır olduğunda en küçük kareler yöntemi ile aynı sonucu verir. Sırt regresyon katsayılarının tahminleri mutlak değerce çok büyük olma eğilimindedir ve bazılarının yanlış işarete sahip olması bile mümkündür. Tahmin vektörleri dikey açıdan ne kadar saparsa, zorluklarla karşılaşılma ihtimali o kadar artar. k değeri artırılarak bir tahmin yapılırsa varyans önemli ölçüde küçültülmüş olunur. Genel kareler toplamının değişmemesinden nedeniyle en küçük kareler yöntemine göre, sırt regresyonunun R^2 ve F değerinde küçük bir azalış meydana gelir.

Sırt regresyon modeline ait k değerinin belirlenmesi öz değerlere dayanır. Sırt regresyon işleminin hangi noktada durağanlaştığını ya da öz değer 1'e en yakın olduğu noktayı belirlemek için Sırt iz grafiğine bakarak ya da k parametresinin değerinin belirlenmesi ile tespit edilebilir. Birçok çalışmada k parametresini belirleyebilmek amacıyla farklı formüller önermişlerdir. Bu formüller arasında öz değeri esas olarak önerilen k için koşul indeksinden yararlanılan Eş. (3.8) aşağıda gösterilmektedir [54]:

$$k \leq \frac{\lambda_{max} - 100\lambda_{min}}{99} \qquad k \neq 0 \tag{3.8}$$

3.3.5. Lasso regresyon

Genel olarak doğrusal regresyon birden fazla değişken içeren veri kümelerine uygulanır. Bu bazı sorunlara neden olmaktadır. Bunlardan en önemlisi değişken sayısı arttıkça, modelin aşırı öğrenme ihtimalinin artmasıdır. Bu durum yerine üç değişken içeren bir model olması daha kullanışlı olabilir. Düzenlileştirme, aşırı öğrenme (overfitting) problemini çözmek için kullanılan bir tekniktir. Lasso regresyonu, doğrusal regresyonun düzenlileştirme tekniklerinden biridir [55].

Lasso regresyon, uzun bir pratik başarı geçmişi olan mevcut verimli algoritmalar ve geniş bir teorik alt yapısı ile nedir ve yüksek boyutlu regresyon problemleri için önemli bir yöntemdir [56]. Lasso (en küçük mutlak büzülme ve operatör seçimi) [57]:

$$\sum_{i=1}^{p} |\beta_i| \leq t \tag{3.9}$$

Eş. 3.9 koşulu ile;

$$\beta^{min}(y - x\beta)'(y - x\beta) \tag{3.10}$$

Şeklinde ifade edilmiştir. Eş. 3.9'da yer alan t parametre değeri, tahminlere için gerçekleştirilen büzülme düzeyini kontrol eder. Lasso en küçük kareler tahmin edicisi $\hat{\beta}_{EKK}$'yı sıfıra büzebilir veya bir kısım i değerleri için $\hat{\beta}_i = 0$ olabilir. Lasso regresyon tekniğinden küçük kareler tekniğine benzer şekilde β katsayılarının belirlenmesi amacıyla kullanılan yanlılık sabitinin de modele eklenmesi ile meydana gelen eşitlik aşağıdaki şekilde gösterilir:

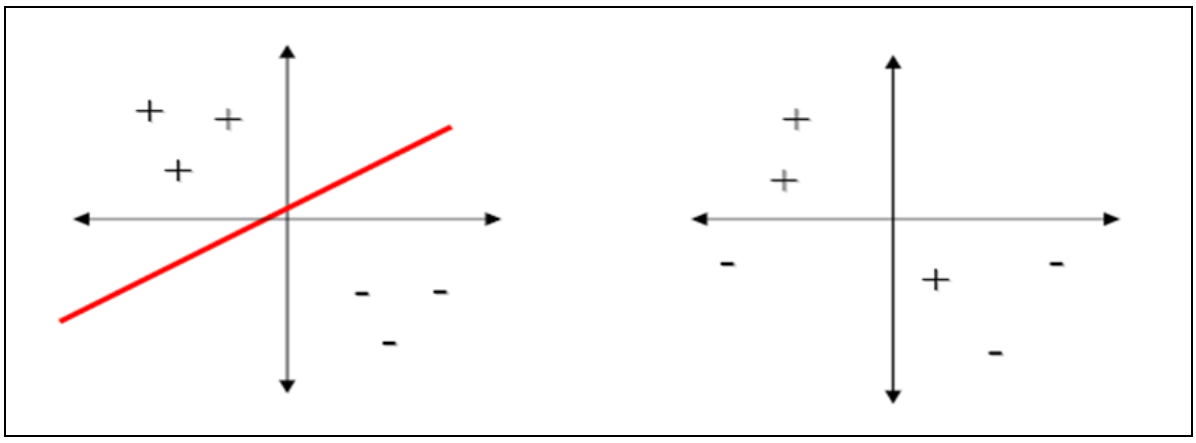
$$\hat{\beta}_{lasso} = \text{argmin}\left\{\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{i=1}^{p} x_{ij}\beta_j\right)^2 + \lambda\sum_{j=1}^{p}|\beta_j|\right\} \tag{3.11}$$

Eş. 3.11'de $l_1 = \sum_{j=1}^{p} |\beta_j|$ ceza fonksiyonudur. Bu fonksiyon pozitif bir değer olup büzülme miktarını denetler.

3.3.6. Destek vektör regresyon

Destek vektör makineleri (DVM), bir makine öğrenimi tekniği olup sınıflandırma ve regresyon modelleri olarak ikiye ayrılır [58]. Bu yöntem diğer klasik makine öğrenimi teknikleriyle karşılaştırıldığında doğrusal olmayan problemlere sunduğu çözüm performansı açısından çok daha etkilidir. Genel olarak sınıflandırma problemlerini çözmek maksadıyla tercih edilen DVM'nin, regresyon için uygulanması Smola ve ark., tarafından 2004 yılında literatüre kazandırılmıştır [59]. Bu yeni teknik ise destek vektör regresyonu (DVR) olarak isimlendirilmiştir [60]. DVM'ler de iki farklı durum söz konusu olabilir. Bunlar verilerin doğrusal ayrılabilen ve doğrusal ayrılamayan bir yapıda olmasıdır.

Doğrusal olarak ayrılabilme durumunda, bu iki değerli veriler direkt olarak bir aşırı düzlem ile ayrılabilecektir. Şekil 3.2'de görüldüğü üzere söz konusu aşırı düzleme, ayırıcı aşırı düzlem ismi verilir. DVM'lerin görevi bu aşırı düzlemin iki farklı grupta yer alan örnek gruba aynı mesafede olmasına imkan tanımaktır [61].



Şekil 3.2. Doğrusal ayrılabilen ve doğrusal ayrılamayan veri

Doğrusal ayrılabilme ve ayrılamama koşulunda, doğrusal bir düzlem veriler, iki ayrı gruba ayırır. Bu durum uygulamalarda her zaman geçerli olmayabilir. Yani doğrusal bir düzlem verileri birbirinden ayıramayabilir.

Doğrusal olmayan sınıflandırıcılar, verilerin doğrusal olarak ayrılamadığı durumlarda doğrusal sınıflandırıcı yerine kullanılabilir. Bu anlamda doğrusal olmayan özellik uzayı; $x_i \in R^n$ gözlem vektörünü daha üst dereceden bir uzayda z vektörüne dönüştürerek, doğrusal sınıflandırıcıları bu yeni uzayda elde etmek mümkün olabilir. Bu z vektörünün ait olduğu

özellik uzayı F ile gösterilsin. Bu durumda Ø ifadesi, $R^n \rightarrow R^F$ eşlemesi koşulu ile $z = \emptyset(x)$ Eş. 3.12 ile aşağıdaki gibi gösterilir [62]:

$$x \in R^n \rightarrow z(x) = [a_1, \emptyset_1(x), \dots \dots a_n\emptyset_n(x)]^T \quad \in \quad R^F \tag{3.12}$$

Doğrusal olmayan ayrılabilirlik durumu göz önüne alınacak olursa, orijinal giriş uzayında zaman ve eğitim verileri doğrusal biçimde ayrılamazlar. Böyle durumlarda DVM, doğrusal olmayan haritalama fonksiyonu aracılığı ile orijinal giriş uzayında doğrusal biçimde sınıflandırma yapabileceği yüksek boyutlu nitelik uzayına dönüşür. Bu yöntem ile çekirdek fonksiyonları kullanılarak tekrarlı çarpım değerlerinin hepsinin ayrı ayrı hesaplanarak bulunması yerine direkt çekirdek fonksiyonda değerin yerine yerleştirerek nitelik uzayındaki değerinin hesaplanması gerçekleşir. Bu şekilde yüksek boyutlu bir nitelik uzayı ile uğraşma ihtimali ortadan kalkar. Çekirdek fonksiyonların diğer bir faydası ise eğitim aşamasındaki bir eğitim verisi için fonksiyon kurulup değerler bulunur ve sonra geriye kalan veriler için kalıp değerleri tamamen hazır olduğundan dolayı hesaplama kolaylaşır [63].

3.3.7. Bayes sırt regresyon (Bayesian ridge regression)

Bayes sırt regresyon (BRR), bayes doğrusal regresyonun özel bir durumu olduğundan ve sırt regresyonuna ait olduğundan, sırt regresyonu ve bayes doğrusal regresyonunun tüm özelliklerine sahiptir. Bayes regresyonunun en uygulanabilir çeşitlerinden biri, bayes sırt regresyonudur. Çünkü BRR, regresyon problemlerinin olasılıksal bir modelini tahmin eder [64].

Sırt regresyon, tahminlerine bir derece sapma ekleyerek standart hataları azaltır. Sırt regresyon, çoklu doğrusal regresyonda sıklıkla ortaya çıkan ortak doğrusallık problemini ele almak için kullanılan bir parametre tahmin yöntemidir [50].

Bir örnek kümesi D olarak tanımlanırsa, örnek kümesindeki örneklerin tümü, sabit fakat bilinmeyen bir olasılık yoğunluk fonksiyonu p(x)'den bağımsız olarak çıkarılır. p(x|D) olarak kaydedilen bu örneklere dayalı olarak x'in olasılık dağılımını tahmin etmek gerekir. Bayes tahmininin esası p(x|D), p(x)'e mümkün olduğunca yakın tercih edilmesidir. p(x) bilinmemesine rağmen, yoğunluk dağılımının iki unsuru şekil ve parametredir.

p(x)'in formunun bilindiğini varsaysak bile θ parametresinin değeri bilinmiyor. Burada p(x|θ), aynı zamanda bir koşullu olasılık yoğunluk fonksiyonu olan bayes tahmininin ilk önemli öğesidir. p(x|θ) biçimi bilindiğinden ve θ parametresinin değeri bilinmediğinden burada x bir test örneği olarak kabul edilebilir. Yani koşullu yoğunluk fonksiyonu p(x|θ), özünde x noktasında θ'nin olabilirlik tahminidir. θ parametresinin değeri bilinmediğinden θ rastgele bir değişken olarak kabul edilebilir.

Ardından, eğitim örneklerini gözlemlenmeden önce temsil etmek için bir sonraki olasılık yoğunluk fonksiyonu p(θ) kullanılabilir. Eğitim örneklerini gözlemleyerek, sonraki olasılık yoğunluğu, sonlu olasılık yoğunluk fonksiyonu p(θ|D)'ye dönüştürebiliriz. Sonraki olasılık yoğunluk korelasyonuna göre, p(θ|D)'nin θ'nin gerçek değerine yakın çok önemli bir tepe noktasına sahip olması beklenir. Bu fonksiyon, bayes tahmininin ikinci ana unsuru olan sonraki olasılık yoğunluğudur. Bayes tahmininin temel problemi p(x|D) bayes tahmininin iki önemli unsuru olan p(x|θ) ve p(θ|D) ile bağlantılıdır.

$$p(x|D) = \int p(x,\theta|D)d\theta = \int p(x|\theta,D)p(\theta|D)d\theta \tag{3.13}$$

Yukarıdaki Eş. 3.13'te x test örneğidir, D eğitim setidir ve x ile D seçimi bağımsızdır. Bu nedenle p(x|θ,D) p(x|θ) olarak yazılabilir. Bu nedenle, bayes tahmininin temel problemi aşağıdaki Eş. 3.14 ile ifade edilir:

$$p(x|D) = \int p(x|\theta)p(\theta|D)d\theta \tag{3.14}$$

Burada p(x|θ), x test örneğinde θ'nin olabilirlik tahminidir. p(θ|D) ise mevcut örnek veri setinde θ'nin sonraki olasılığıdır. Sonraki olasılık p(θ|D) aşağıdaki Eş. 3.15'te gösterilmiştir:

$$p(\theta|D) = \frac{p(D|\theta)p(\theta)}{P(D)} = \frac{p(D|\theta)p(\theta)}{\int p(D|\theta)p(\theta)d\theta} \tag{3.15}$$

$$p(D|\theta) = \prod_{k=1}^{n} p(x_k|\theta) \tag{3.16}$$

D örnek kümesinde n tane örnek olduğunu açıkça ifade etmek için aşağıdaki Eş. 3.17'de belirtilen notasyon kullanılmıştır.

$$D^n = \{x_1, x_2, \dots \dots x_n\} \tag{3.17}$$

Önceki formüle göre, n>1 olduğunda aşağıdaki Eş. 3.18'de söz konusu olur:

$$D^n = \{x_1, x_2, \dots \dots x_n\} \tag{3.18}$$

$$p(D^n|\theta) = p(x_n|\theta)p(D^{n-1}|\theta) \tag{3.19}$$

Eş. 3.19'a göre aşağıdaki eşitlikler kolayca elde edilir:

$$p(\theta|D^n) = \frac{p(x_n|\theta)p(D^{n-1}|\theta)p(\theta)}{\int p(x_n|\theta)p(D^{n-1}|\theta)p(\theta)d\theta} \tag{3.20}$$

$$p(\theta|D^n) = \frac{p(x_n|\theta)p(\theta|D^{n-1})}{\int p(x_n|\theta)p(\theta|D^{n-1})d\theta} \tag{3.21}$$

Gözlenen örnek olmadığında, $p(\theta|D_0) = p(\theta)$, θ parametresinin ilk tahmini olarak tanımlanır. Ortaya çıkan bu model bayes sırt regresyon (BRR) olarak adlandırılır [65].

3.3.8. Rastgele orman regresyon (Random forest regression)

Rastgele orman regresyon (RFR), kendi başlarına regresyon işlevi gören yüzlerce hatta binlerce karar ağacı üretir ve RFR'nin nihai çıktısı, tüm karar ağaçlarının çıktılarının ortalamasıdır. Sınıflandırma ve regresyon ağacı (CART) olarak da adlandırılan bir karar ağacı, ilk olarak Breiman ve tarafından diğerleri 1984 yılında tanıtılan istatistiksel bir modeldir [66].

Karar ağacı parametrik olmayan bir modeldir. Sınıf yoğunlukları için herhangi bir ön parametre almaz ve ağaç yapısını düzeltmez. Ağaç, beslenen verilerin karmaşıklığına bağlı olarak öğrenme sürecinde büyür [67]. Her karar ağacı, karar düğümleri ve yaprak düğümlerden oluşur. Karar düğümleri, beslenen her örneği bir test fonksiyonu ile değerlendirir ve örneğin özelliklerine göre farklı dallara iletir.

x, m özellik içeren girdi vektörünü temsil etmek üzere $x = \{x_1,\ x_2, \dots \dots x_m\}$ y ve S_n çıktı değerleri gözlemden oluşan eğitim veri kümesi aşağıdaki Eş. 3.22 ile gösterilir:

$$S_n = \{(x_1, y_1), (x_2, y_2), \dots \dots (x_n, y_n)\}, x \in \mathbb{R}^n, y \in \mathbb{R} \quad (3.22)$$

Verinin eğitilmesi sırasında, giriş verileri algoritma tarafından her bir düğümde bölünür, böylece bölünmüş fonksiyonların parametreleri S_n seti ile optimum hale gelir. Karar ağacı, ilk adımda tüm değişkenler arasında en iyi ayrımı yapma stratejisi ile çalışır. Bu bölme işlemi kök kısımda başlar ve her düğüm ile yeni x girdisine kendi bölme işlevini uygular. Bu işlem, bir uç düğüme (ağaç yaprakları) bağlanana kadar yinelemeli olarak tekrarlanır. Maksimum sayıda düzeye ulaşıldığında veya bir düğüm önceden tanımlanmış sayıdan daha az gözlem içerdiğinde ağaç durdurulur. Bu eğitim sürecinin sonunda S_n üzerinde bir $\hat{h}(x, S_n)$ tahmin fonksiyonu oluşturulur.

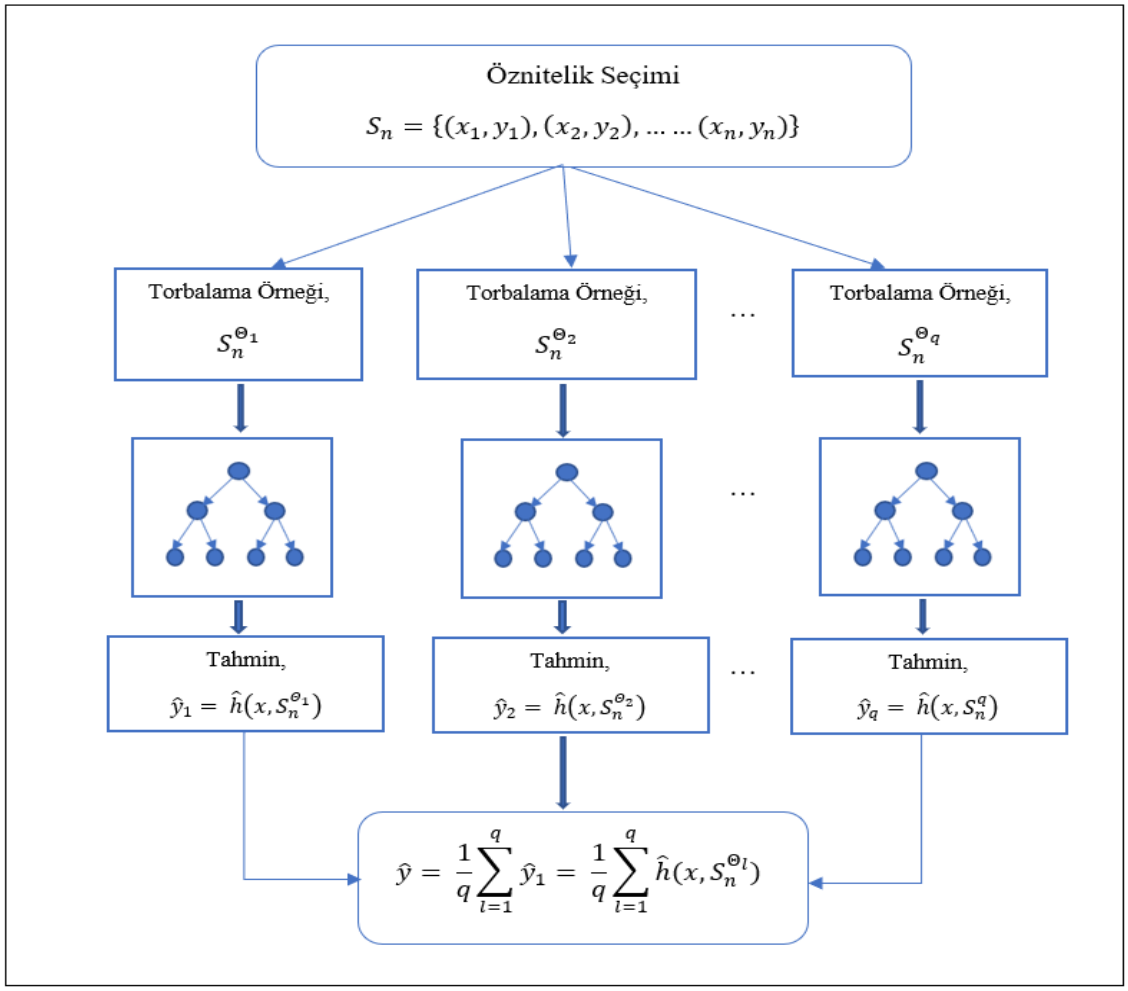
Rastgele orman regresyon modeli, CART karar ağacı algoritmasının bir uzantısıdır ve daha iyi bir tahmin performansı sunar. RFR'nin eğitim aşaması, birbiriyle ilişkili birden fazla karar ağacı oluşturmaktır. RFR'deki her ağaç, rastgele bir tahmin edici alt kümesiyle büyütülür ve bu nedenle 'rastgele' orman olarak adlandırılır. RFR, L ağaç yapılı temel sınıflandırıcı $h(x, \Theta_k)$ yı kullanır, burada k = 1,2,…..L, Θ_k bağımsız ve özdeş olarak dağıtılmış rastgele vektörlerin bir ailesidir, X ise bir girdi vektörüdür [66].

RFR oluşturulan tüm karar ağaçlarını 'torbalama (önyükleme toplama)' adı verilen bir algoritma kullanarak birleştiren bir topluluk yöntemidir. Torbalama, Breiman [68] tarafından önerilen bir tekniktir ve tahminle ilişkili varyansı azaltmak için birçok regresyon yöntemiyle birlikte kullanılabilir ve böylece tahmin performansını iyileştirir. RFR, her karar ağacı için bir özellik alt kümesini rastgele örnekleyerek ve/veya her karar ağacı için bir eğitim verisi alt kümesini rastgele örnekleyerek oluşturulabilir. Örneklerin rastgele toplandığı bu işleme "Torbalama (Bootstrap)" denir ve her gözlemin seçilme olasılığı 1/n olan S_n içerisinden değiştirilerek n gözlem rastgele seçilir bir torbalama örneği elde edilir. Torbalama algoritması birkaç torbalama örneği ($S_n^{\Theta_1} \dots \dots S_q^{\Theta_q}$) seçer ve q tahmin ağaçlarının bir koleksiyonunu oluşturmak için önceki ağaç karar algoritmasını bu örneklere $\hat{h}(x, S_n^{\Theta_1}), \dots \dots \dots, \hat{h}\left(x, S_n^{\Theta_q}\right)$ uygular. Topluluk (ensemble) metodu, her bir ağaca karşılık gelen $\hat{y}_1 = \hat{h}(x, S_n^{\Theta_1}), \dots \dots \dots, \hat{y}_q = \hat{h}\left(x, S_n^{\Theta_q}\right)$ çıktıları üretir. Daha sonra tüm ağaçların

çıktılarının ortalaması alınarak toplama gerçekleştirilir. Sonuç olarak, çıktının y tahmini şu şekilde elde edilir:

$$\hat{y} = \frac{1}{q}\sum_{l=1}^{q}\hat{y}_1 = \frac{1}{q}\sum_{l=1}^{q}\hat{h}(x, S_n^{\Theta_l}) \tag{3.23}$$

Tahmin modellerinden RFR regresyon yapısı aşağıdaki Şekil 3.3'te gösterilmektedir [69].



Şekil 3.3. Rastgele orman regresyon yapısı

Torbalama kullanmanın belirgin bir avantajı, farklı ağaçların korelasyonundan kaçınarak ağaçların çeşitliliğini sağlamak, RFR'de oluşturulan farklı eğitim veri alt kümelerinden buna göre yenilerini elde edebilmektir [69]. Bazı veriler eğitimde birden fazla kullanılabilirken bazıları hiç kullanılmayabilir. Böylece, giriş verilerinde küçük değişikliklerle

karşılaşıldığında RFR regresyonunu daha sağlam hale getiren torbalama kullanımı nedeniyle daha fazla RFR kararlılığı elde edilir [66].

Torbalamanın diğer bir avantajı, farklı eğitim örnekleri aracılığıyla ilişkisiz ağaçlar oluşturduğu için gürültüye karşı bağışıklık oluşturmasıdır. Zayıf bir tahmin edici gürültüye duyarlı olabilir ancak birkaç yapıyla ilişkili karar ağacının ortalaması gürültü duyarlılığını büyük ölçüde azaltabilir [70]. RFR'nin önemli bir özelliği, içindeki ağaçların budamadan büyümesi ve bu da onları hesaplama açısından hafif hale getirmesidir [69].

RFR, ayarlanması gereken yalnızca iki parametre ile fazlasıyla kullanıcı dostu bir algoritmadır. Bu parametreler; ağaç sayısı (n_{tree}) ve her bölme için rastgele özellik sayısı (m_{try}) ormanda oluşturulmaya çalışılır [71]. Bu nedenle mükemmel performans elde etmek için parametrelerde ince ayar yapmaya çok az ihtiyaç duyar. Genel olarak, ormanda ne kadar çok ağaç yetiştirilirse, o kadar sağlam ve yüksek tahmin doğruluğu elde edilebilir. Bununla birlikte, artan miktarda ağaç, gelişmiş bir hesaplama yüküne yol açabilir. Genelleme hatası ağaç sayısı arttıkça yakınsar, yani belirli bir noktaya ulaştıktan sonra tahmin doğruluğu artırılamaz. Varsayılan değer olan n_{tree} = 500'ün genellikle tahmin için kullanıldığı bu yakınsamaya izin vermek için ağaç sayısının yeterince yüksek ayarlanması gerekir. Diğer parametre m_{try} ise model gücünü belirleyen ve her ağacın gücünü ve ormandaki herhangi iki ağaç arasındaki ilişkiyi tanımlayan hassas bir parametredir. Artan m_{try} her ağacın gücünü artırabilir, ancak ağaçlar arasındaki korelasyon aynı zamanda artacaktır [72].

Çizelge 3.1. Rastgele orman regresyon mimarisi

Adım 1. Orijinal verilerden n_{tree} torbalama örnekleri çizilir, bir torbalama alt kümesi, orijinal veri kümesinin öğelerinin yaklaşık 2/3'ünü içerir.
Adım 2. Torbalama örneğini kullanarak budanmamış bir regresyon ağacını büyütmek için; her düğümde, tüm tahmin ediciler arasında en iyi bölmeyi seçmek yerine, tahmin edicilerin m_{try}'sini rastgele örnekler ve bu değişkenler arasından en iyi bölmeyi seçer.
Adım 3. n_{tree} ağaçlarının tahminlerini toplayarak yeni verileri tahmin eder (yani sınıflandırma için çoğunluk oyları ve regresyon için ortalama).

İyileştirilmiş ağaç gücü, model performansını artırırken, ağaçlar arasındaki artan korelasyon onu zayıflatabilir. Tüm öngörücü değişkenlerin sayısının üçte biri olan m_{try}'nin varsayılan değerinin genellikle iyi bir seçim olduğu bildirilmiştir [71]. Özetlemek gerekirse, RFR'yi büyütmek için temel adımlar Çizelge 3.1'de listelenmiştir.

Her bir regresyon ağaç yapısı için, orijinal verilerden değiştirilerek yeni bir eğitim seti (torbalama örnekleri) oluşturulur. Bu nedenle, torbalama işleminde q. ağacın eğitimi için tüm örnekler seçilmez ve eğitim verilerinin bir kısmı eğitim örneğinde tekrar tekrar kullanılabilir. Seçilmeyen numuneler, torba dışı numuneler (OOB (Out-of-bag)) adı verilen başka bir alt kümenin parçası olarak dahil edilir. Normalde yeni eğitim örneklerinin üçte ikisi regresyon fonksiyonunu oluşturmak için kullanılırken üçte biri OOB örneğini oluşturur. Eğitim örneğiyle bir regresyon ağacı oluşturulduğunda, regresyon ağacının performansını değerlendirmek için OOB örneği kullanılır. Bu bir tür yerleşik çapraz doğrulama işlemidir. Bu şekilde RFR, ağaçlar eğitim sırasında bu gözlemleri görmediği için harici bir veri alt kümesi kullanmadan genelleme hatasının tarafsız bir tahminini hesaplayabilir.

Ayrıca, aşırı öğrenme (overfitting) tehlikesi RFR kullanılarak büyük ölçüde azaltılabilir. Bu yerleşik doğrulama özelliği, RFR'nin genelleme yeteneğini geliştirir. Toplam öğrenme hatasını elde etmek için, OOB örneğini kullanan her bir ağacın tahmin hatasının ortalaması Eş. 3.24 ile elde edilebilir [66]:

$$\mathrm{MSE} \approx \ \mathrm{MSE}^{\mathrm{OOB}} = \ \mathrm{n}^{-1} \sum_{\mathrm{i}=1}^{\mathrm{n}} [\hat{\mathrm{y}}(\mathrm{x_i}) - \mathrm{y_i}]^2 \tag{3.24}$$

Burada $\hat{y}(x_i)$, belirli bir girdiye karşılık gelen tahmini çıktıdır örnek x_i oysa y_i gerçek değerleri temsil eden gözlemlenen çıktıdır ve n, OOB örneklerinin toplam sayısıdır. Bu hata, bilinmeyen örneklere uygulandığında RFR tahmininin ne kadar verimli olduğunu belirleyebilir. OOB hatası, genelleme hatasının tarafsız bir tahminidir ve RFR'nin üretim hatasının sınırlayıcı bir değerini ürettiği kanıtlanmıştır.

3.4. Model Performans Değerlendirme Ölçütleri

Regresyon modelleri sonucu elde edilen hedef değerin tahmin tahmin performansını değerlendirmek amacıyla çeşitli yöntemler kullanılır. Bunlardan bazıları aşağıda açıklanmıştır.

3.4.1. Determinasyon katsayısı (R^2)

Determinasyon katsayısı, bir model ile tahmin edilen değerlerin gözlemlenen gerçek değerlerle ne kadar yakından eşleştiğinin bir ölçüsüdür. Bir model için ideal R^2 değeri 1'dir. Bu değer 1 olduğunda modelin hedef sınıfın tüm değişkenliğini tam olarak açıklayabildiği anlamına gelir. R^2 değeri aşağıdaki Eş. 3.25'te gösterilmiştir [73]:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2} \tag{3.25}$$

Eş. 3.25'te yer alan y_i gözlem değeri, $\hat{y}_i$ tahmin edilen değer, $\bar{y}_i$ gözlem değerlerinin ortalamasını ifade eder.

R^2=0, olduğunda girdi ve çıktı değişkenleri arasında korelasyon yoktur. R^2=1 ise girdi ve çıktı değişkenleri arasında güçlü bir pozitif ilişki varken; R^2=-1 ise girdi ve çıktı değişkenleri arasında ters yönlü bir ilişki vardır.

3.4.2. Hata kareleri ortalaması (MSE)

MSE bir regresyon eğrisinin bir dizi noktaya ne kadar yakın olduğunu ifade eder. MSE, pozitif değer alır. MSE değeri sıfıra yakınsa bu modelin iyi bir tahmin yaptığını gösterir [74].

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3.26}$$

Eş. 3.26'da yer alan y_i gözlem değeri, $\hat{y}_i$ tahmin edilen değer, n örneklem sayısını ifade eder.

3.4.3. Ortalama mutlak hata (MAE)

MAE, zaman serisi ve regresyon modellerinde doğruluğu değerlendirmek için sıkça kullanılmaktadır. Bu yöntem, gerçek değerler ile veriye en iyi uyan çizgi arasındaki ortalama dikey uzaklıktır [75].

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{n} \tag{3.27}$$

Eş. 3.27'de yer alan y_i gözlem değeri, $\hat{y}_i$ tahmin edilen değer, n örneklem sayısını ifade eder.

3.4.4. Ortalama hata kareleri karekökü (RMSE)

RMSE bir tahmin modelinin tahmin ettiği değer ile gerçek değer arasındaki mesafenin büyüklüğünü ölçen metriktir. RMSE tahmin hatalarının standart sapmasıdır ve değerinin sıfır olması modelin hiç hata yapmadığı anlamına gelmektedir. Bu ölçütün avantajı, yüksek hataları daha fazla cezalandırmasıdır. Bu sebeple diğer yöntemlere göre daha sık tercih edilir [75].

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n}} \tag{3.28}$$

Eş. 3.28'de yer alan y_i gözlem değeri, $\hat{y}_i$ tahmin edilen değer, n örneklem sayısını ifade eder.

4. DENEYSEL ÇALIŞMA

Bu çalışmada, üç farklı makine öğrenimi modeli ile 1 Temmuz 2020 ile 23 Kasım 2020'ye kadar onaylanmış günlük COVID-19 vaka sayıları kullanılarak Türkiye'nin 12 istatiski bölgesi için ayrı ayrı salgın tahmin modelleri oluşturulmuştur. Bölgelerin durumunun ülkenin genel tahmin durumuyla karşılaştırılması amacıyla Türkiye geneli içinde ayrıca tahmin modelleri kurulmuştur. COVID-19'un yayılım eğilimini göstermek ve yetkililerin etkili sağlık politikaları üretmesine yardımcı olmak amacıyla tahminler sunulmuştur. Rastgele orman regresyon (RFR), bayes sırt regresyon (BRR) ve lineer regresyon (LR) analizleri, Python yazılım paketinin veri madenciliği kütüphaneleri kullanılarak veri setlerine uygulanmıştır.

4.1. Bölgeler Hakkında Genel Bilgiler

Çalışmada kullanılan veri seti Türkiye'nin 12 istatistiki bölgesinin günlük COVID-19 verilerinden oluşmaktadır. Bu resmi veri seti Sağlık Bakanlığı internet sitesinden elde edilerek Kaggle platformu üzerinden paylaşılan günlük COVID-19 veri setidir [76].

Haziran 1941 tarihinde yapılan Birinci Coğrafya Kongresi, Türkiye'yi 7 coğrafi bölge olarak ayırmışlardır. 2000'li yıllarda ise Avrupa Birliği (AB)'ne tam üyeliğin gerçekleştirilmesi için gereken bir kriter olarak İstatistiki Bölge Birimleri Sınıflandırması (İBBS) sistemi uygulamaya konmuştur. Bu sisteme göre ülke genelinde veya bölgesel olarak gerçekleşecek planlamaların bu bölgeler göz önüne alınarak yapılmasıdır. AB, İBBS bölgelerinin oluşturulmasını Kalkınma Ajansları'nın kurulabilmesi için ön şart olarak mecburi tutmuştur. Türkiye 22 Eylül 2002 tarih ve 24884 sayılı resmî gazetede yayınlanan Bakanlar Kurulu kararı ile İBBS bölgelerine ayrılmıştır [77]. İBBS sistemi, 3 esas göz önüne alınarak oluşturulmuştur. Bunlardan birincisi, ülkelerin mevcut bölge yapısının İBBS sınıflandırmasına temel oluşturmasıdır. İkinci esas ise benzer potansiyele sahip ve coğrafik olarak yakın alanların sınıflandırılması yoluyla bölgelerin oluşturulmasıdır. Devlet Planlama Teşkilatı, Devlet İstatistik Enstitüsü ve İçişleri Bakanlığı yetkililerinden oluşan bir komisyon, İBBS için görevlendirilmiştir. Bu komisyon, AB ülkelerindeki yapıya benzeyen ve 3 farklı düzeyden oluşan bir sistem oluşturmuştur. Çizelgede gösterildiği üzere Türkiye İBSS bölgeleri adı altında, Düzey 1'de 12, Düzey 2'de 26 ve Düzey 3'te 81 il olarak gruplandırılmıştır.

Çizelge 4.1. Türkiye İBBS bölgeleri

BÜYÜK İSTATİSTİK BÖLGELERİ DÜZEY 1	ALT İSTATİSTİK BÖLGELERİ DÜZEY 2	ALT İSTATİSTİK BÖLGELERİ DÜZEY 3
İSTANBUL	İSTANBUL	İstanbul
BATI MARMARA	TEKİRDAĞ	Tekirdağ, Edirne, Kırklareli
	BALIKESİR	Balıkesir, Çanakkale
EGE	İZMİR	İzmir
	AYDIN	Aydın, Denizli, Muğla
	MANİSA	Manisa, Afyon, Kütahya, Uşak
DOĞU MARMARA	BURSA	Bursa, Eskişehir, Bilecik
	KOCAELİ	Kocaeli, Sakarya, Düzce, Bolu, Yalova
BATI ANADOLU	ANKARA	Ankara
	KONYA	Konya, Karaman
AKDENİZ	ANTALYA	Antalya, Isparta, Burdur
	ADANA	Adana, Mersin
	HATAY	Hatay, Kahramanmaraş, Osmaniye
ORTA ANADOLU	KIRIKKALE	Kırıkkale, Aksaray, Niğde, Nevşehir, Kırşehir
	KAYSERİ	Kayseri, Sivas, Yozgat
BATI KARADENİZ	ZONGULDAK	Zonguldak, Karabük, Bartın
	KASTAMONU	Kastamonu, Çankırı, Sinop
	SAMSUN	Samsun, Tokat, Çorum, Amasya

Çizelge 4.1. (devamı) Türkiye İBBS bölgeleri

DOĞU KARADENİZ	TRABZON	Trabzon, Ordu, Giresun, Rize, Artvin, Gümüşhane
KUZEYDOĞU ANADOLU	ERZURUM	Erzurum, Erzincan, Bayburt
	AĞRI	Ağrı, Kars, Iğdır, Ardahan
ORTADOĞU ANADOLU	MALATYA	Malatya, Elazığ, Bingöl, Tunceli
	VAN	Van, Muş, Bitlis, Hakkari
GÜNEYDOĞU ANADOLU	GAZİANTEP	Gaziantep, Adıyaman, Kilis
	ŞANLIURFA	Şanlıurfa, Diyarbakır
	MARDİN	Mardin, Batman, Siirt, Şırnak

İBBS bölgelerinin belirlenmesinde mevcut coğrafi bölgeler dikkate alınması gerekirken Türkiye'de daha farklı kriterlere bağlı olarak bölgeler sınıflandırılmıştır. Başta illerin nüfus yoğunluğu olmak üzere kültürel yapı ve illerin gelişmişlik durumu göz önüne alınmıştır [77].

Harita 4.1. Türkiye'nin İBBS düzey 1 bölgeleri [78]

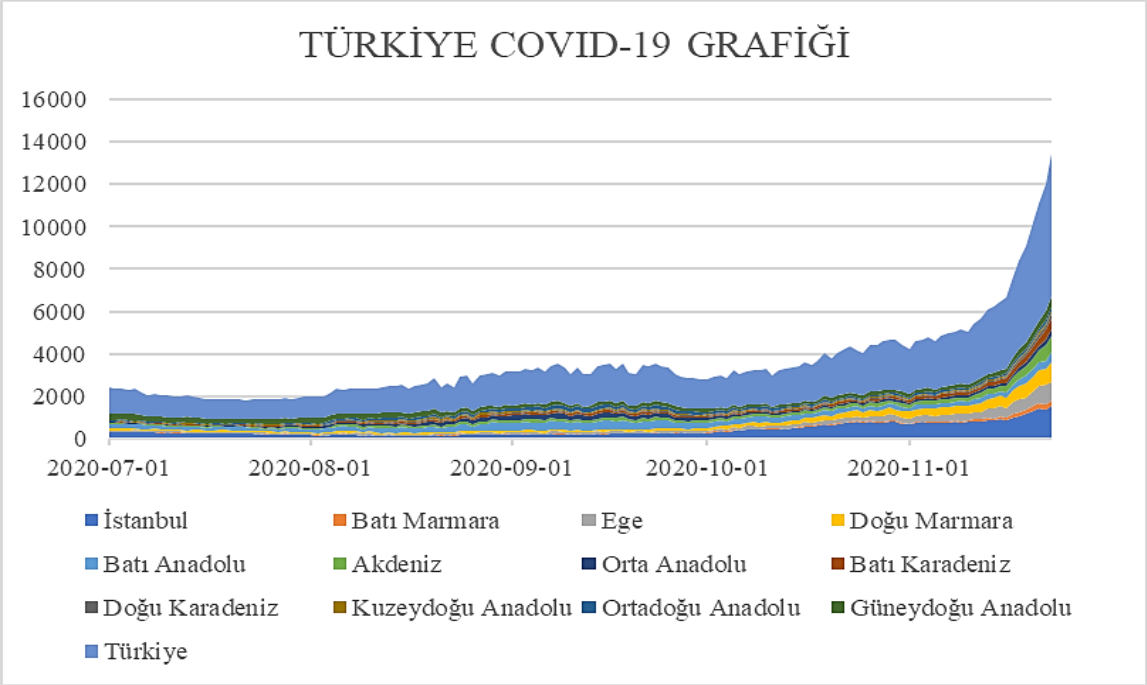
4.2. COVID-19 Veri Seti Tanımlayıcı İstatistiksel Bilgileri

Veri setinin bir bölümü aşağıda Çizelge 4.2'de gösterilmektedir.

Çizelge 4.2. İBSS COVID-19 günlük verileri

Tarih	Blg 1	Blg 2	Blg 3	Blg 4	Blg 5	Blg 6	Blg 7	Blg 8	Blg 9	Blg 10	Blg 11	Blg 12
01/07/20	297	7	84	115	183	68	22	22	10	15	48	327
02/07/20	288	6	89	116	184	66	22	21	9	16	46	324
03/07/20	279	16	72	89	229	48	26	30	12	47	46	283
04/07/20	268	4	68	135	170	51	31	31	12	28	48	313
05/07/20	311	6	86	105	132	57	27	20	8	22	37	350
06/07/20	272	12	61	78	240	69	28	29	11	10	35	250
08/07/20	266	5	75	83	154	68	22	22	8	18	43	250
09/07/20	310	4	64	92	125	60	29	29	15	18	37	281
10/07/20	261	7	64	106	153	63	28	34	15	14	36	249
11/07/20	265	4	71	87	140	51	32	30	12	18	40	255
12/07/20	262	9	58	75	134	60	36	46	10	9	41	252
13/07/20	256	10	63	101	118	63	40	29	6	33	53	210
14/07/20	238	8	62	79	157	50	39	31	13	28	38	270
15/07/20	251	8	55	81	148	52	40	33	12	21	30	250
16/07/20	249	6	51	86	140	58	26	22	19	31	41	221
17/07/20	235	6	54	72	141	54	29	29	15	26	39	219
18/07/20	239	8	62	70	147	59	29	21	22	24	40	204
19/18/20	242	9	51	87	136	55	33	24	23	27	27	207
20/07/20	245	6	63	80	122	57	35	14	16	25	27	230

Çizelge 4.2'de yer alan Blg1: İstanbul, Blg2: Batı Marmara, Blg3: Ege, Blg4: Doğu Marmara, Blg5: Batı Anadolu, Blg6: Akdeniz, Blg7: Orta Anadolu, Blg8: Batı Karadeniz, Blg9: Doğu Karadeniz, Blg10: Kuzeydoğu Anadolu, Blg11: Ortadoğu Anadolu, Blg12: Güneydoğu Anadolu bölgelerini temsil etmektedir.



Şekil 4.1. İBBS COVID-19 grafiği

Şekil 4.1'de COVID-19 salgınına ait günlük vaka sayılarının bölgesel olarak değişimi gösterilmektedir. Grafikte günlük vaka sayılarının küçük dalgalanmalar ile sürekli artarak dört aylık süreçte her bölge için en yüksek değerlere ulaştığı görülmektedir.

Bu çalışmada kullanılan veriler, 1 Temmuz 2020 ile 23 Kasım 2020 tarihleri arasındaki onaylanmış günlük COVID-19 verileridir. Yalnızca laboratuvar testleri ile pozitif olduğu doğrulanan hastalar, resmi olarak doğrulanmış vakalar olarak kabul edilir ve analizler için doğrulanmış veriler kullanılmıştır. Veriler, kurulan modeli eğitmek için bir eğitim veri kümesine (%70) ve modeli doğrulamak için bir test veri kümesine (%30) bölünmüştür. Söz konusu veri setleri için Python programı kullanılarak hazırlanan tanımlayıcı istatistiki bilgiler veri setinin tamamını içermektedir ve aşağıda sunulmuştur.

İstatistiksel ölçümler, veri setinin merkezi eğilimini, dağılımını ve şeklini özetleyen tanımlayıcı istatistikler üretir. Sayısal değişkenler için gün sayısı, ortalama, yüzdelikler gibi

bazı istatistiksel değerler hesaplanır. Bu istatistikler ve analizlerine, veri biliminde yaygın bir uygulama olan tanımlayıcı istatistikler denir.

Çizelge 4.3. Blg-1, Blg-2 ve Blg-3 tanımlayıcı istatistikleri

İstanbul		Batı Marmara		Ege	
gün sayısı	146,000000	gün sayısı	146,000000	gün sayısı	146,000000
ortalama	387,513699	ortalama	3,609589	ortalama	160,746575
standart sapma	304,010418	standart sapma	47,628356	standart sapma	158,263500
min	99,000000	min	4,000000	min	45,000000
25%	177,500000	25%	9,000000	25%	73,000000
50%	243,500000	50%	12,500000	50%	99,500000
75%	596,250000	75%	28,000000	75%	181,000000
max	1557,000000	max	262,000000	max	902,000000

Çizelge 4.3'te belirtilen gün sayısı, ilgili sütunun satırında kaç adet veri olduğunu gösterir. Ortalama, sütundaki verilerin ortalama değeridir. Standart sapma, veri kümesine ait ortalama değer ve değişken arasındaki fark olarak tanımlanabilir. Düşük bir standart sapma, veri noktalarının ortalamaya daha yakın olduğunu gösterir.

Yüzdelik dilimler (quartiles: %25, %50, %75), verilerin belirli bir yüzdesinden daha düşük olan değeri tanımlayanlar. Min ve max, ifadeleri ise veri seti içindeki en küçük değer ile en büyük değeri gösterir.

Çizelge 4.4. Blg-4, Blg-5 ve Blg-6 tanımlayıcı istatistikleri

Doğu Marmara		Batı Anadolu		Akdeniz	
gün sayısı	146,000000	gün sayısı	146,000000	gün sayısı	146,000000
ortalama	175,417808	ortalama	248,239726	ortalama	136,424658
standart sapma	162,633404	standart sapma	87,127913	standart sapma	103,778134
min	55,000000	min	96,000000	min	48,000000
25%	81,000000	25%	191,250000	25%	83,000000
50%	97,500000	50%	230,500000	50%	115,000000
75%	212,500000	75%	295,250000	75%	148,000000
max	899,000000	max	533,000000	max	728,000000

Çizelge 4.5. Blg-7, Blg-8 ve Blg-9 tanımlayıcı istatistikleri

Orta Anadolu		Batı Karadeniz		Doğu Karadeniz	
gün sayısı	146,000000	gün sayısı	146,000000	gün sayısı	146,000000
ortalama	118,342466	ortalama	91,184932	ortalama	40,684932
standart sapma	65,906877	standart sapma	77,934544	standart sapma	30,308472
min	22,000000	min	14,000000	min	6,000000
25%	75,250000	25%	53,500000	25%	27,000000
50%	102,000000	50%	81,500000	50%	38,000000
75%	166,750000	75%	100,000000	75%	44,000000
max	294,000000	max	551,000000	max	226,000000

Çizelge 4.6. Blg-10, Blg-11 ve Blg-12 tanımlayıcı istatistikleri

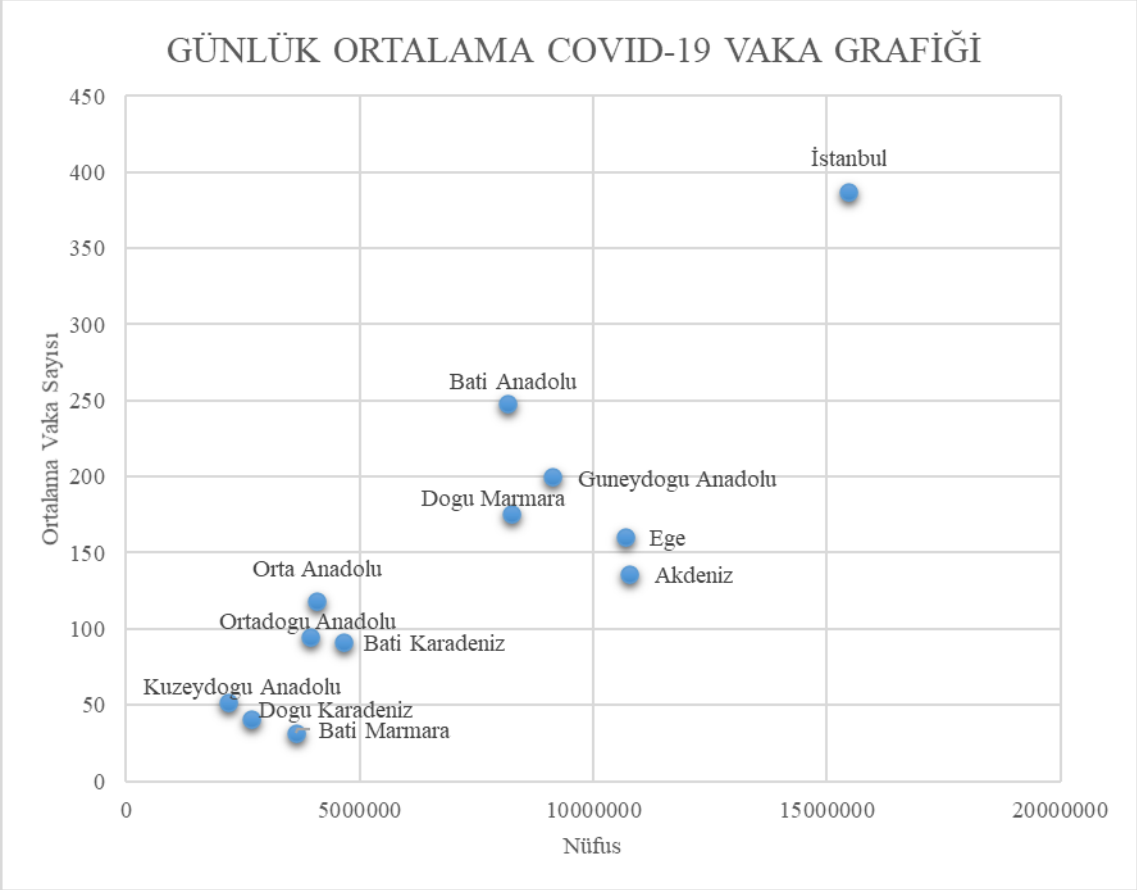
Kuzeydoğu Anadolu		Ortadoğu Anadolu		Güneydoğu Anadolu	
gün sayısı	146,000000	gün sayısı	146,000000	gün sayısı	146,000000
ortalama	52,157534	ortalama	94,595890	ortalama	199,856164
standart sapma	20,369920	standart sapma	42,014214	standart sapma	62.886322
min	9,000000	min	27,000000	min	110,000000
25%	39,250000	25%	67,250000	25%	150,250000
50%	50,500000	50%	92,000000	50%	184,500000
75%	65,750000	75%	120,750000	75%	240,500000
max	137,000000	max	197,000000	max	468,000000

Çizelge 4.7. Bölgelerin nüfus sayısı

Bölge Adı	Nüfus
İstanbul	15 462 452
Batı Marmara	3 632 398
Ege	10 689 115
Doğu Marmara	8 235 816
Batı Anadolu	8 168 261
Akdeniz	10 759 218
Orta Anadolu	4 088 228
Batı Karadeniz	4 638 622
Doğu Karadeniz	2 677 584
Kuzeydoğu Anadolu	2 192 453
Ortadoğu Anadolu	3 951 286
Güneydoğu Anadolu	9 118 921

Tanımlayıcı istatistik Çizelge 4.3'te görüldüğü üzere ortalama COVID-19 vaka sayısı en yüksek olan değer İstanbul bölgesine ait iken en düşük değer Batı Marmara bölgesine aittir. İstanbul'un nüfusu; 15 milyon 462 bin 452 iken Batı Marmara bölgesinin toplam nüfusu; 3 milyon 632 bin 398'dir. İstanbul'da nüfusun yoğun olması salgının yayılımı için elverişli bir ortam oluşturmuştur. İstanbul için tanımlayıcı istatistik değerlerine bakacak olursak; veri setinin 146 satırdan oluştuğunu, vaka sayısının ortalama değerinin 387,513699 olduğu, vaka

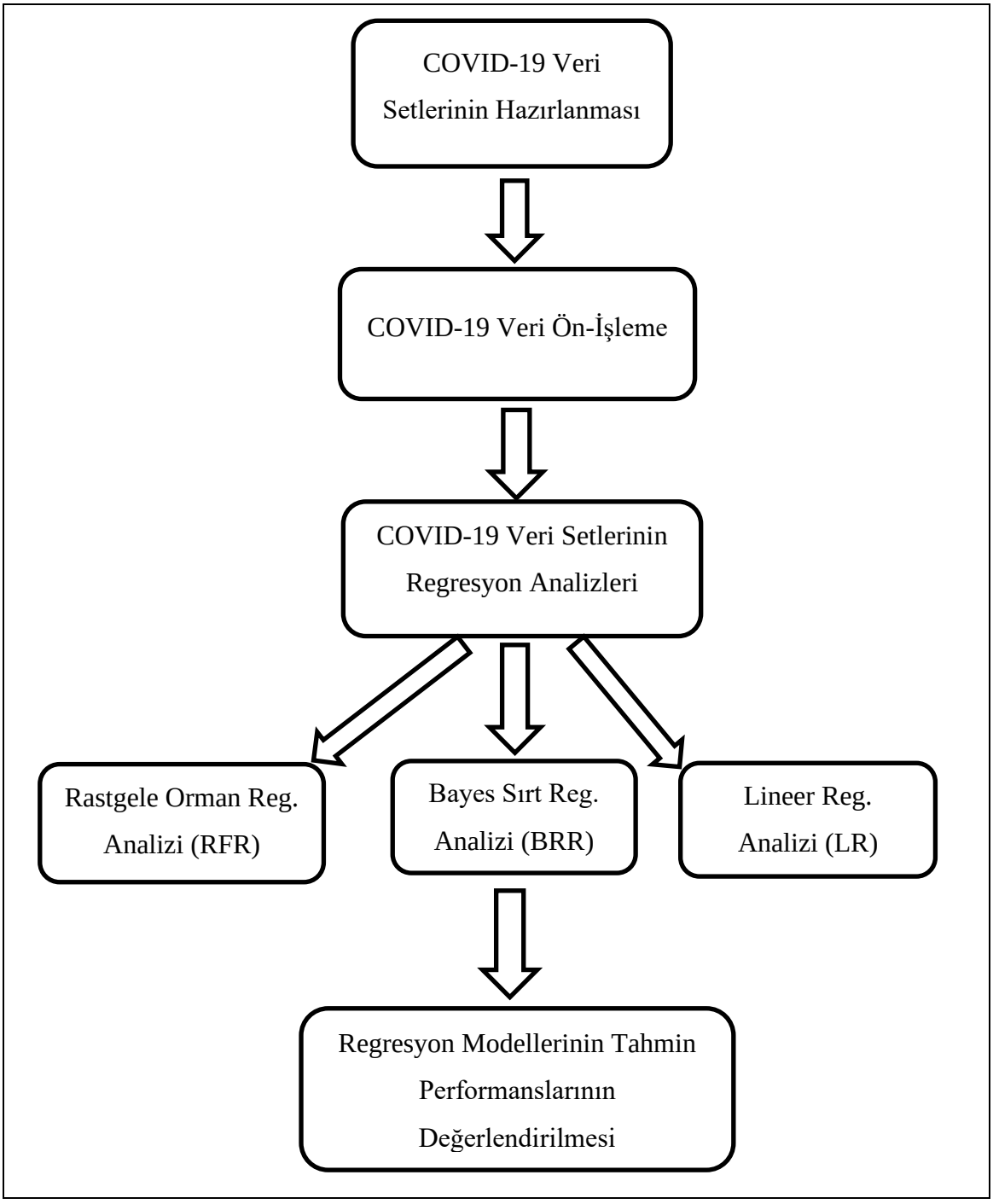
sayının en az olduğu gün 99, en fazla olduğu gün 1557 kişinin COVID-19 olduğu ve veri setinin standart sapmasının 304,010418 olduğu görülür. İstanbul için yüzdelik dilimlere bakacak olursak; ilk çeyrek (%25)'te, verilerin dörtte biri 177,5'in altında kalmıştır. Medyan (%50), tüm verilerin yarısı altında olmak üzere veri setinin tam ortasında 243,5 değeri bulunur. Üçüncü çeyrek (%75), verilerin dörtte üçünün altında kaldığı nokta ise 596,25'tir.



Şekil 4.2. İBBS günlük ortalama COVID-19 vaka sayısı ve nüfus karşılaştırması

Şekil 4.2'de ise bölgelerin nüfus yoğunlukları ile günlük ortalama COVID-19 vaka sayısı karşılaştırılmalı olarak verilmiştir. Kuzey Doğu Anadolu ve Doğu Karadeniz gibi nüfus yoğunluğu düşük olan bölgelerin vaka sayısının diğer kalabalık bölgelere oranla daha az olduğu açıkça görülmektedir.

4.3. Deneysel Çalışmanın Yöntemi



Şekil 4.3. Deneysel çalışma akış şeması

Deneysel çalışmaların akış diyagramı Şekil 4.3'te verilmektedir. Burada ilk adım verilerin hazırlanması aşamasıdır. Bu aşamada her bölgenin veri setleri veri ön işleme süreci ile analizler için hazır hale getirilmiştir.

4.4. Makine Öğrenimi Modellerinin Uygulanması

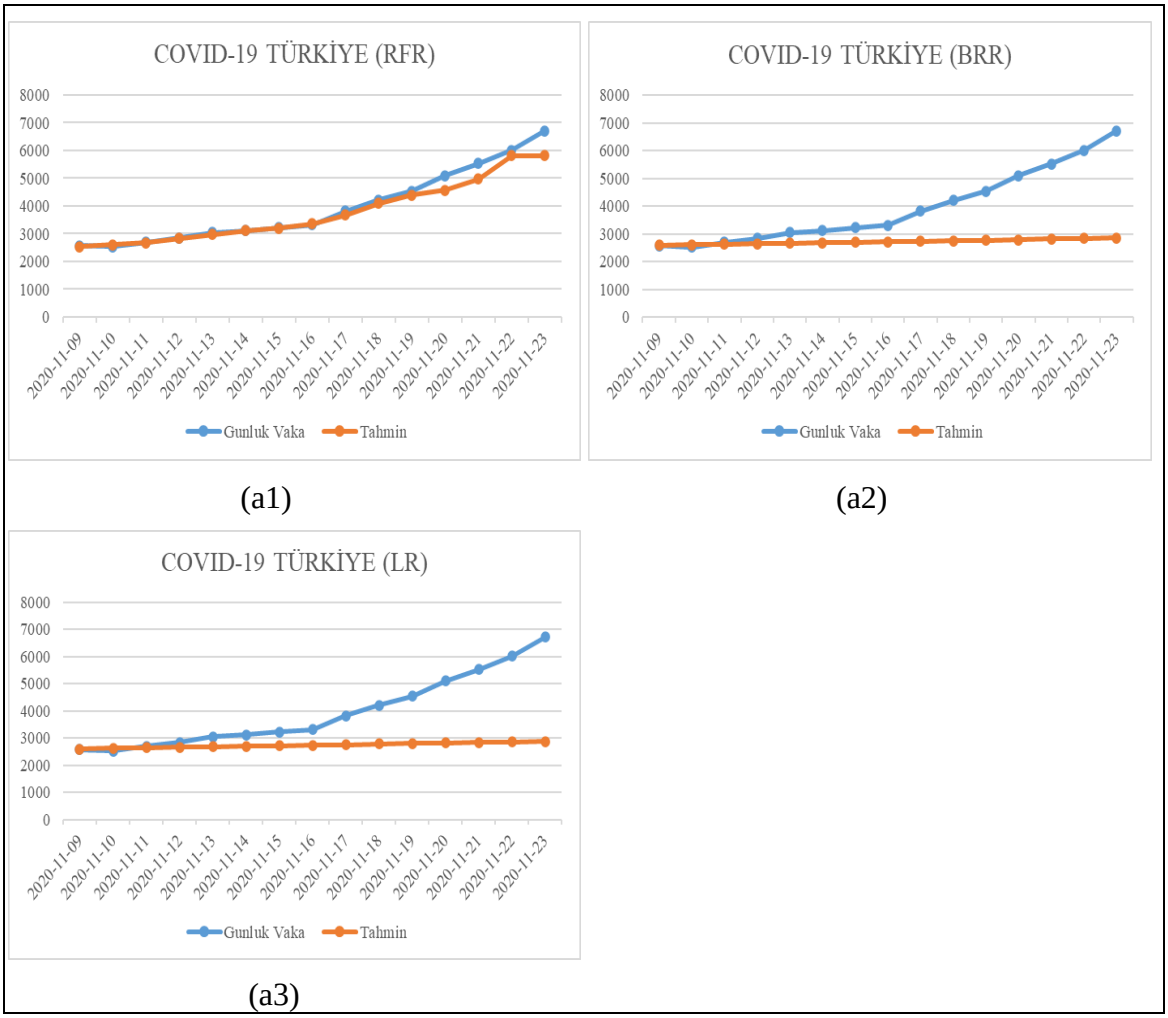
Bu çalışmada, 1 Temmuz 2020 ile 23 Kasım 2020'ye kadar doğrulanmış günlük COVID-19 vaka sayıları üç farklı regresyon yöntemi kullanılarak modellenmiştir. Rastgele orman regresyonu (RFR), bayes sırt regresyon (BRR) ve lineer regresyon (LR) analizleri Python yazılım paketinin veri madenciliği kütüphaneleri kullanılarak gerçekleştirilmiştir.

Bu çalışmanın esas amacı, Türkiye için bölgeler bazında detaylı bir analiz gerçekleştirerek regresyon algoritmalarının salgın hastalıkların tahmini konusunda performanslarının etkili olduğunu göstermektir. Öncelikli olarak Türkiye geneli ile ilgili modeller oluşturulmuştur. Türkiye geneli oluşturulan regresyon modellerinin amacı ise 12 istatistiki bölgenin tahminlerinin genel tahminden nasıl farklılaştığını göstererek detaylı analizlerin daha gerçekçi olduğunu ifade etmektir.

Tahmin modellerinin uygulanmasında Python programından yararlanılmıştır. Her bölgeye ait 1 Temmuz 2020 – 23 Kasım 2020 tarihleri arasındaki 146 günlük verinin %70'i eğitim, %30'u test verisi olarak belirlenerek modeller kurulmuş ve son 15 gün için COVID-19 vaka sayıları tahmin edilmiştir. RFR, BRR ve LR tahmin modelleri her bölgeye ayrı ayrı uygulanmıştır.

RFR, BRR ve LR algoritmaları için Python uygulamasının "Scikit-learn" kütüphanesinden sırasıyla "RandomForestRegressor", "BayesianRidge" ve "LinearRegression" fonksiyonları kullanılmıştır. Python uygulamasında RFR, BRR ve LR modelleri için yazılan kodlar detaylı olarak EK-1, EK-2, EK-3 ve EK-4'te paylaşılmıştır.

Türkiye'de 9 Kasım 2020- 23 Kasım 2020 tarihleri arasındaki her bölge için doğrulanmış 15 günlük gerçek COVID-19 vaka sayıları ile RFR, BRR ve LR sonucu elde edilen tahminler aşağıdaki şekillerde karşılaştırmalı olarak gösterilmiştir. Oluşturulan tahmin modelleri determinasyon katsayısı (R^2), ortalama hata kareleri karekökü (RMSE), hata kareleri ortalaması (MSE), ortalama mutlak hata (MAE) performans ölçütleri ile değerlendirilmiştir.

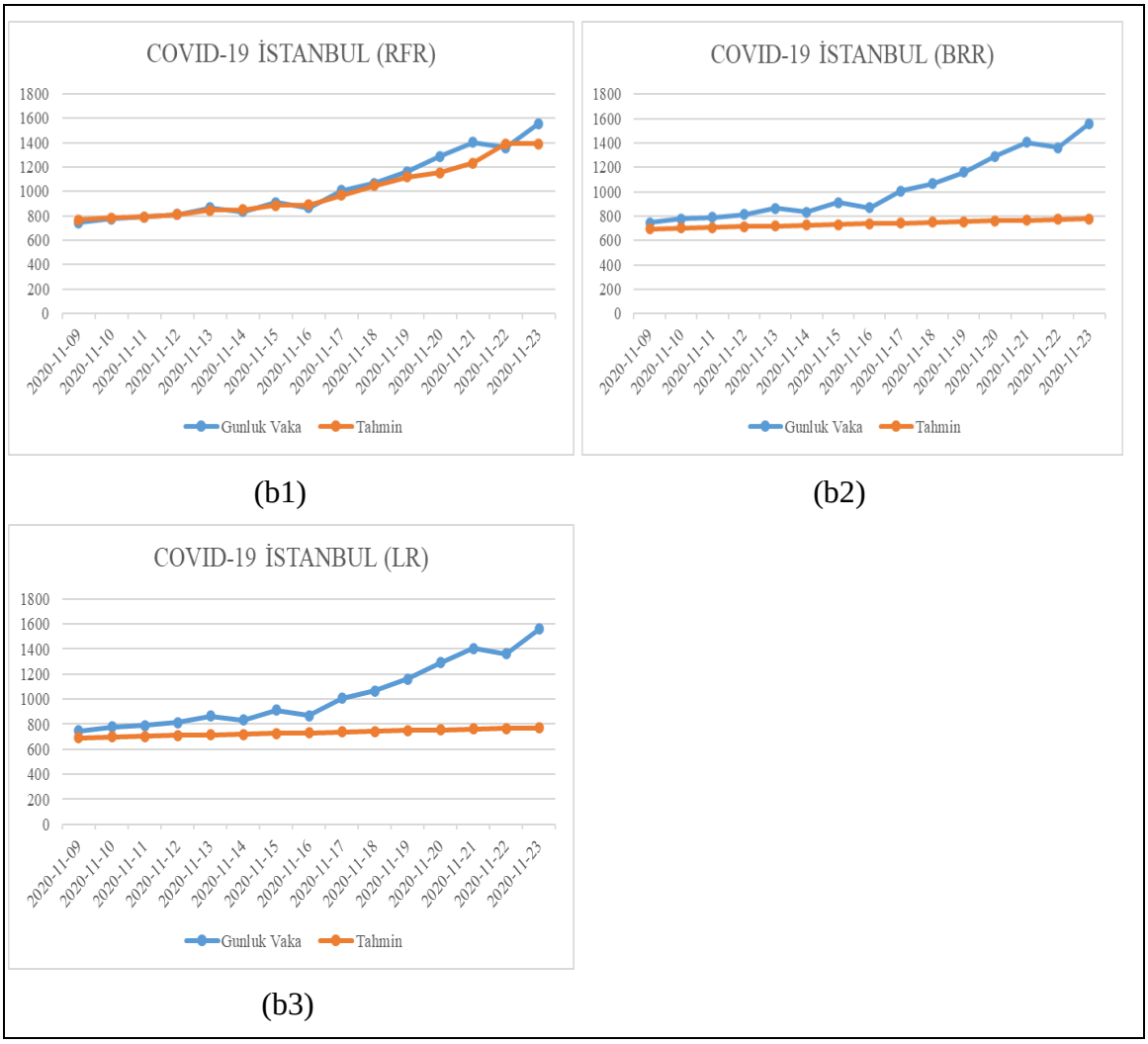


Şekil 4.4. Türkiye geneli a1: RFR; a2: BRR; a3:LR için gerçek ve tahmin değerleri

Çizelge 4.8. Türkiye geneli için tahmin performans değerleri

Türkiye	R^2	RMSE	MSE	MAE
RFR	0,98	140,82	19831,24	81,27
BRR	0,52	768,11	589988,2	367,82
LR	0,52	771,4	595055,92	383,67

Şekil 4.4'te Türkiye'de vaka sayılarının kısa sürede hızlı bir artış gösterdiği görülmektedir. Çizelge 4.8'de genel tahmin modeli sonuçları gösterilmektedir. RFR algoritması R^2=0,98 değeri ile çok etkili bir tahmin gerçekleştirmiştir. BRR ve LR algoritmaları R^2=0,52 değeri ile düşük tahmin performansı göstermiştir.

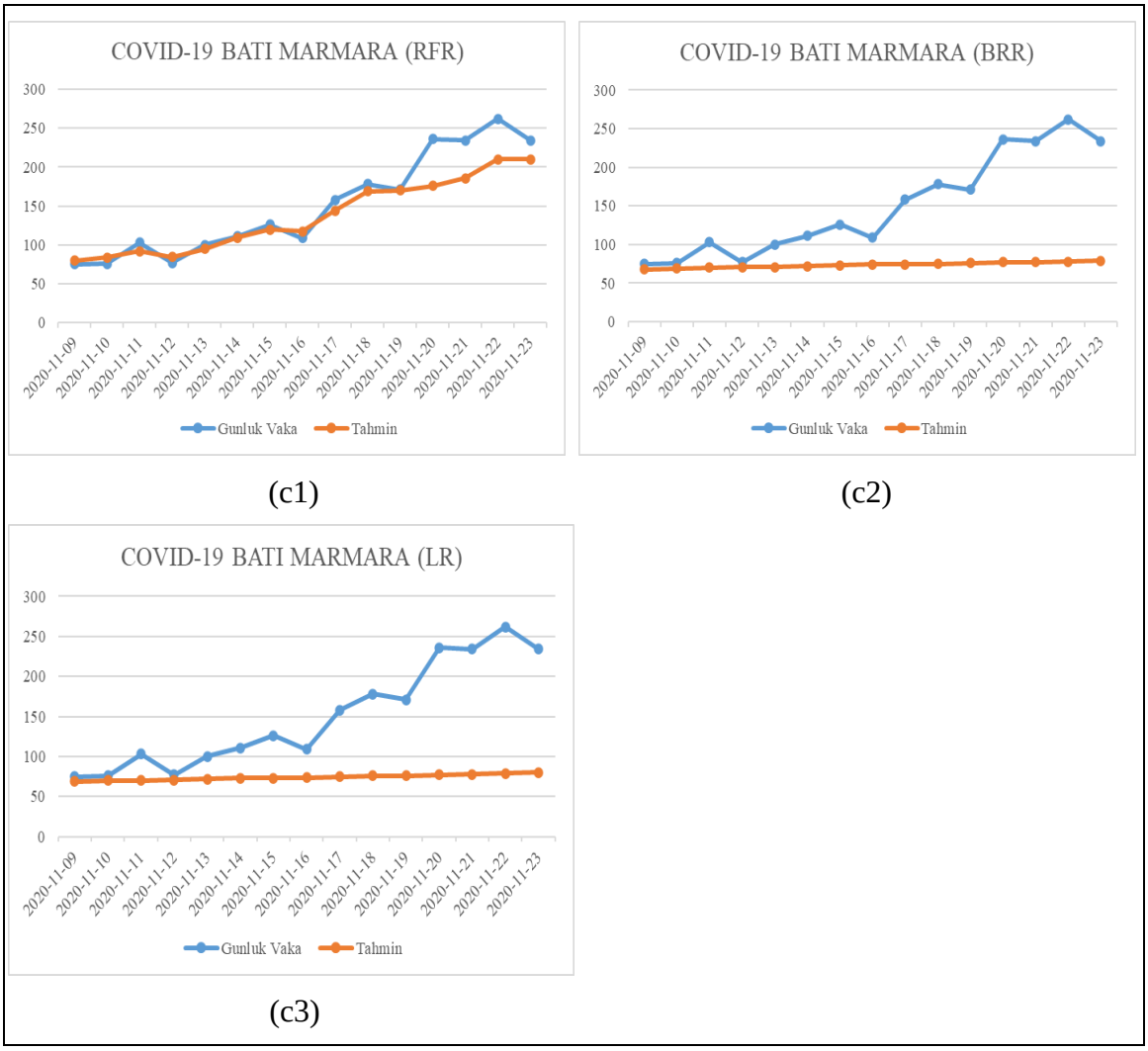


Şekil 4.5. İstanbul b1: RFR; b2: BRR; b3:LR için gerçek ve tahmin değerleri

Çizelge 4.9. İstanbul için tahmin performans değerleri

İstanbul	R^2	RMSE	MSE	MAE
RFR	0,98	43,13	1860,23	29,67
BRR	0,57	215,97	46642,84	168,59
LR	0,69	181,32	32876,48	113,56

Şekil 4.5'te görüldüğü gibi İstanbul bölgesinde COVID-19 vaka sayısı en yüksek seviyededir. Bu durumun temel nedeni nüfus yoğunluğunun fazla olmasının salgınının yayılmasında etkili olmasıdır. Çizelge 4.9'da RFR algoritmasının R^2=0,98 değeri ve diğer ölçütler açısından belirgin bir farkla iki tahmin modelini de geride bıraktığı görülmektedir.

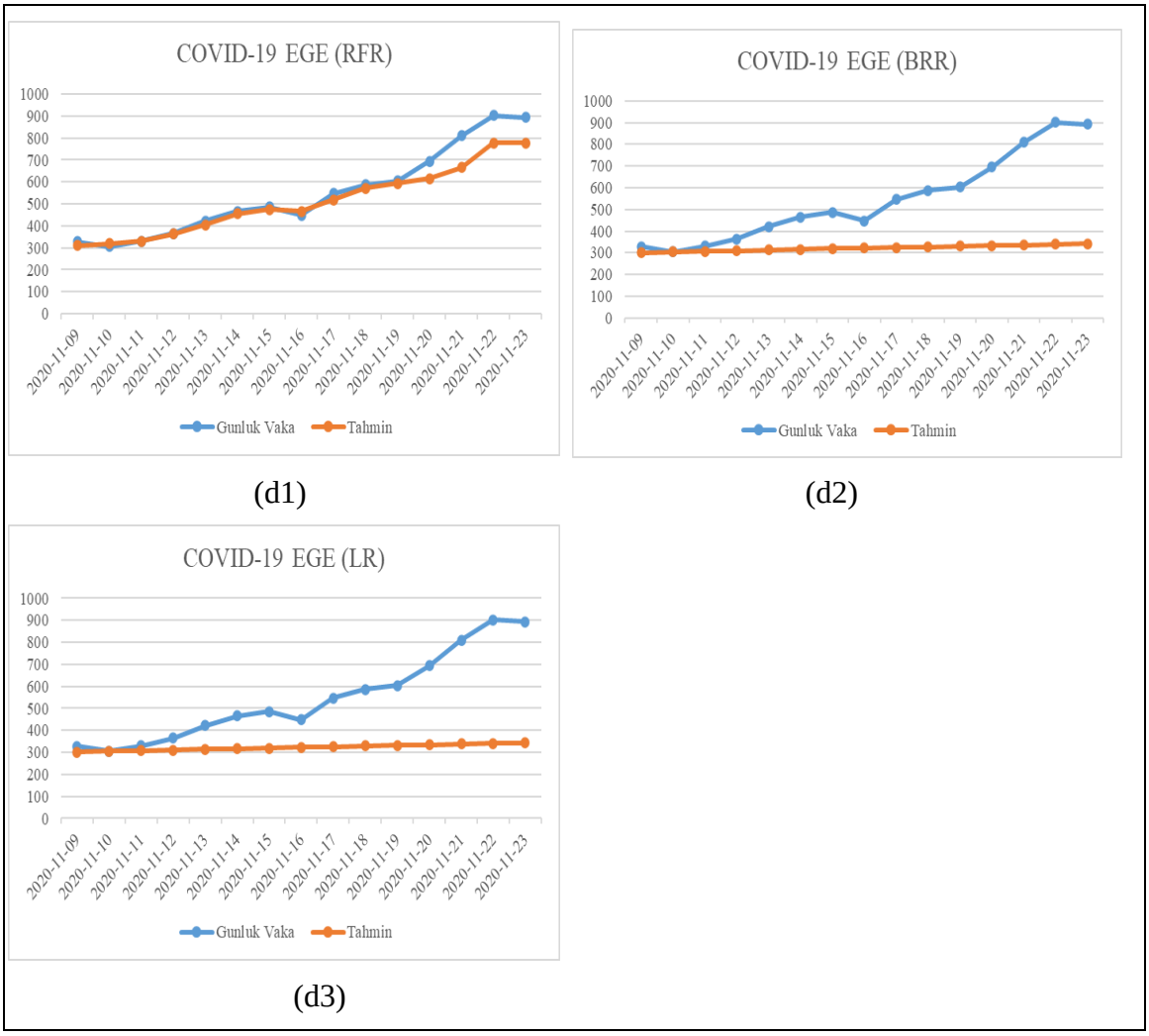


Şekil 4.6. Batı Marmara c1: RFR; c2: BRR; c3:LR için gerçek ve tahmin değerleri

Çizelge 4.10. Batı Marmara için tahmin performans değerleri

Batı Marmara	R^2	RMSE	MSE	MAE
RFR	0,94	14,54	211,28	6,62
BRR	0,37	46,92	2201,3	25,52
LR	0,38	46,53	2165,16	24,7

Şekil 4.6'da görüldüğü gibi Batı Marmara bölgesi gerçek verileri yüksek dalgalanmalı bir yapıya sahiptir. Bu yüksek değişkenlik koşulunda bile Çizelge 4.10'da verilen R^2=0,94 değeri ile RFR algoritması etkili performans göstermiştir. BRR ve LR algoritmaları bütün tahmin performansları açısından RFR'nin gerisinde kalmıştır.



(d1) (d2)

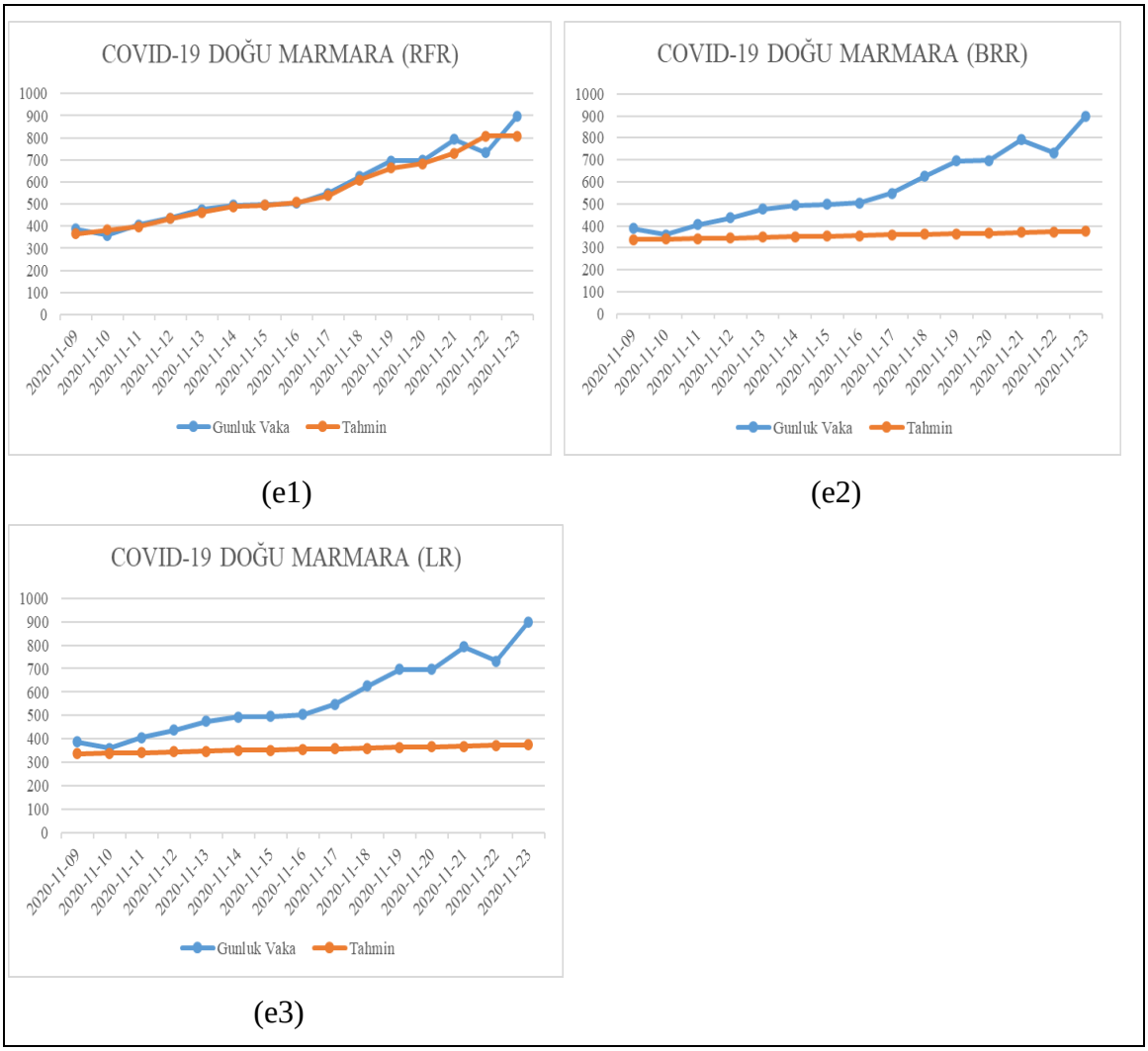
(d3)

Şekil 4.7. Ege d1: RFR; d2: BRR; d3:LR için gerçek ve tahmin değerleri

Çizelge 4.11. Ege için tahmin performans değerleri

Ege	R^2	RMSE	MSE	MAE
RFR	0,96	32,84	1078,43	15,47
BRR	0,47	134,66	18134,03	76,86
LR	0,49	132,84	17645,68	72,71

Şekil 4.7'de Ege bölgesinin kısa sürede hızlı bir artış gösterdiği görülmektedir. BRR ve LR algoritmaları ile RFR arasında belirgin farklılık Çizelge 4.11'de görülmektedir. RFR kısa süreli ve hızlı değişen vaka sayılarını R^2=0,96 ile gerçeğe yakın bir doğrulukla tahmin etmeyi başarmıştır. RMSE=32,84 değeri de diğer iki yöntemden açık ara daha düşüktür.

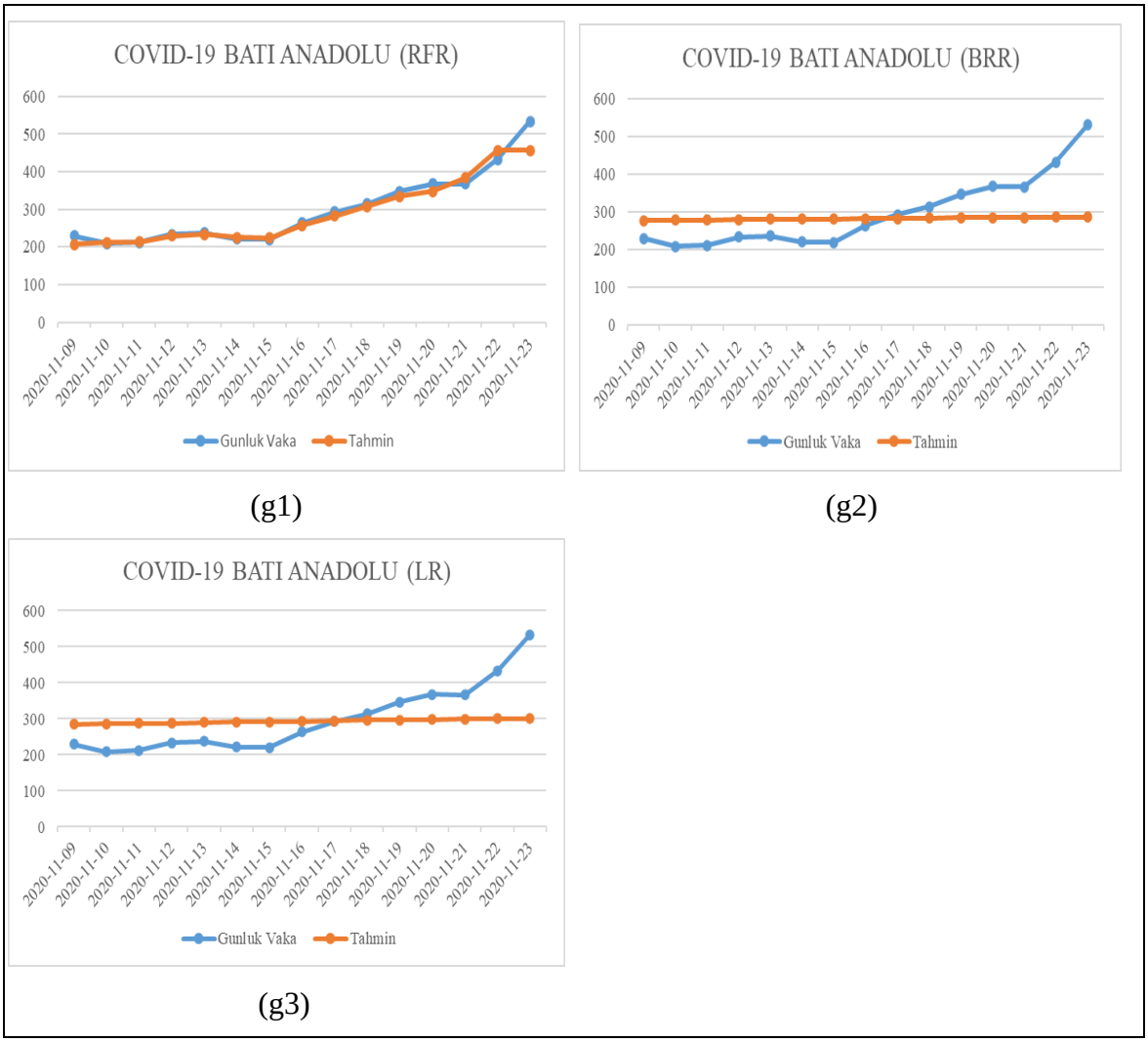


Şekil 4.8. Doğu Marmara e1: RFR; e2: BRR; e3:LR için gerçek ve tahmin değerleri

Çizelge 4.12. Doğu Marmara için tahmin performans değerleri

Doğu Marmara	R^2	RMSE	MSE	MAE
RFR	0,98	19,83	393,12	13,69
BRR	0,54	118,21	13972,45	80,93
LR	0,58	112,41	12636,88	69,99

Şekil 4.8'te Doğu Marmara bölgesinin vaka sayının nispeten doğrusal olarak arttığı görülmektedir. Buna bağlı olarak BRR ve LR algoritmaları, Çizelge 4.12'de görüldüğü üzere R^2=0,50 üzerine çıkmayı başarmış ancak yine de R^2=0,98 değeri ile RFR algoritmasının gerisinde kalmışlardır.

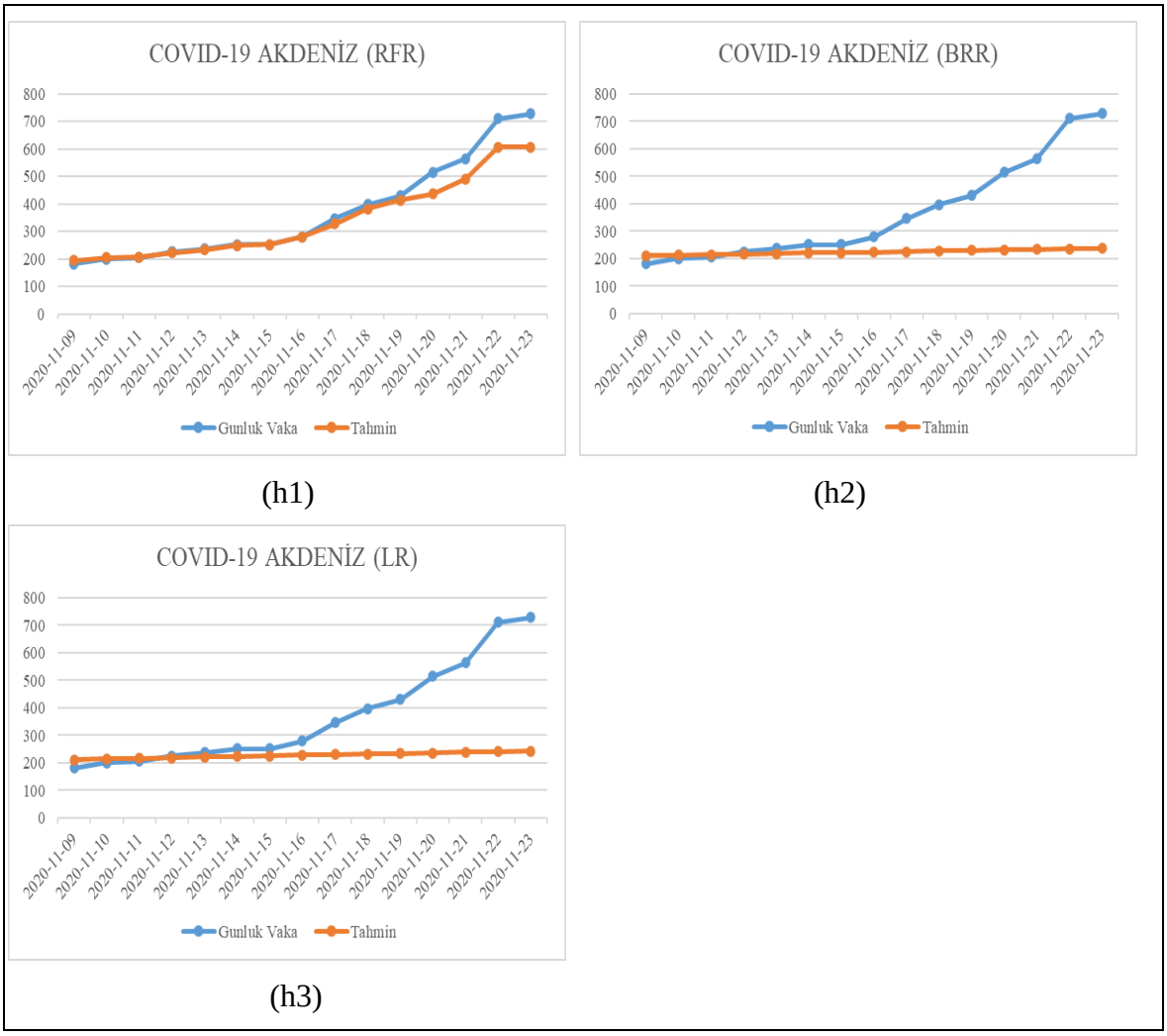


Şekil 4.9. Batı Anadolu g1: RFR; g2: BRR; g3:LR için gerçek ve tahmin değerleri

Çizelge 4.13. Batı Anadolu için tahmin performans değerleri

Batı Anadolu	R^2	RMSE	MSE	MAE
RFR	0,91	28,03	785,42	20,72
BRR	0,17	86,82	7536,85	72,13
LR	0,59	60,29	3635,45	49,79

Şekil 4.9'de Batı Anadolu'ya ait grafikler gösterilmektedir. Çizelge 4.13'te RFR algoritması R^2=0,91 ile başarılı bir tahmin gerçekleştirmiştir. R^2 değeri, ortalama hata kareleri toplamının karekökü olduğundan hata değerindeki en küçük farklılıklar R^2 değerinde büyük çaplı değişiklikler meydana getirir. BRR algoritması LR ile yakın tahminler yapmasına rağmen küçük hata değerleri bile oransal değeri büyük ölçüde farklılaştırmıştır.

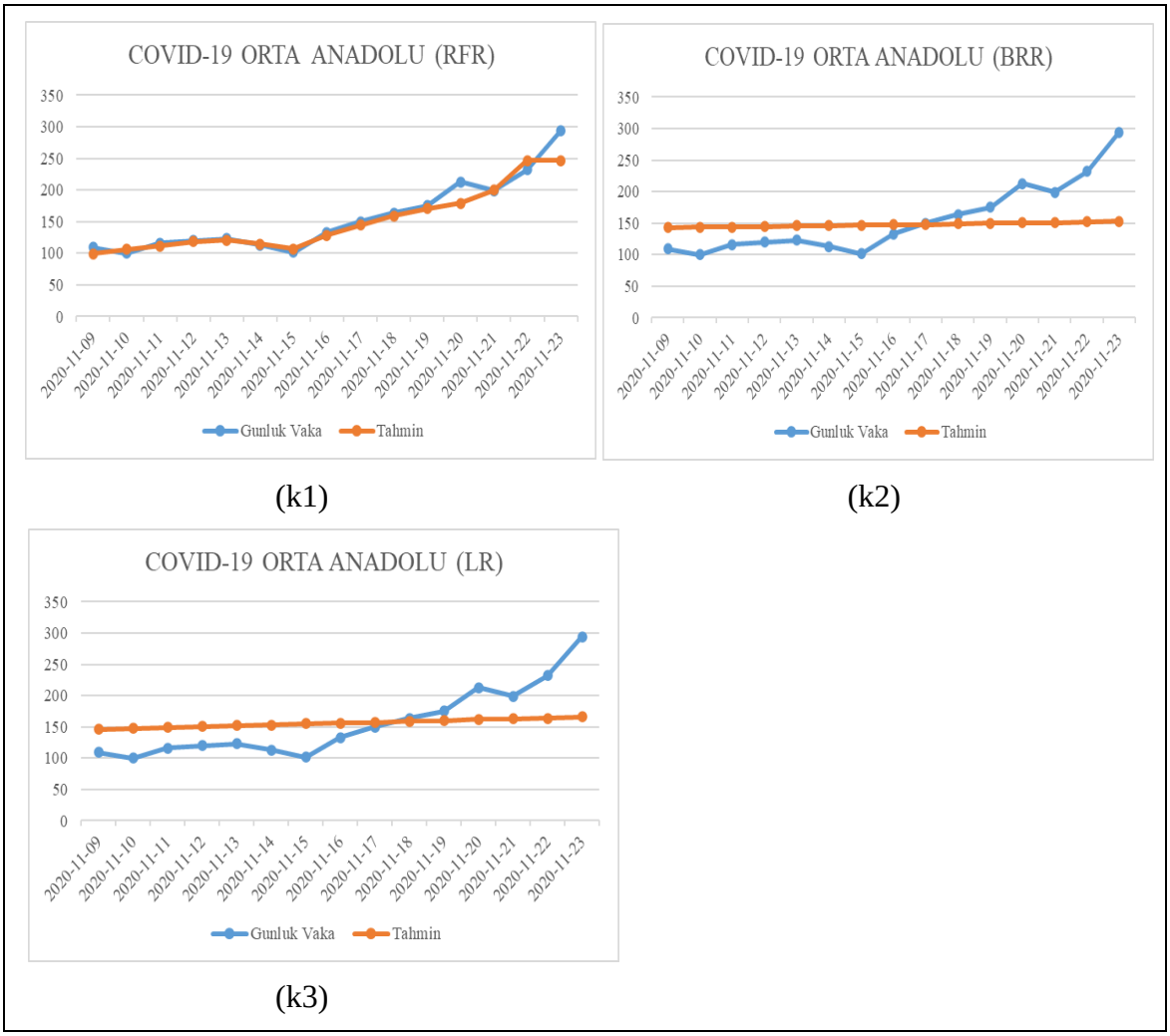


Şekil 4.10. Akdeniz h1: RFR; h2: BRR; h3:LR için gerçek ve tahmin değerleri

Çizelge 4.14. Akdeniz için tahmin performans değerleri

Akdeniz	R^2	RMSE	MSE	MAE
RFR	0,96	26,04	677,94	15,98
BRR	0,39	101,16	10232,53	45,92
LR	0,39	101,07	10216,1	48,05

Şekil 4.10'da Akdeniz bölgesi için gerçek değerler ile tahmin değerleri gösterilmektedir. Çizelge 4.14'te RFR algoritması R^2=0,96 ile diğer iki yöntemden çok daha başarılı sonuçlar ortaya koymuştur. RFR yöntemi bu bölgede de RMSE=26,04 değeri ile diğer yöntemlere göre en küçük hata değerine sahiptir.

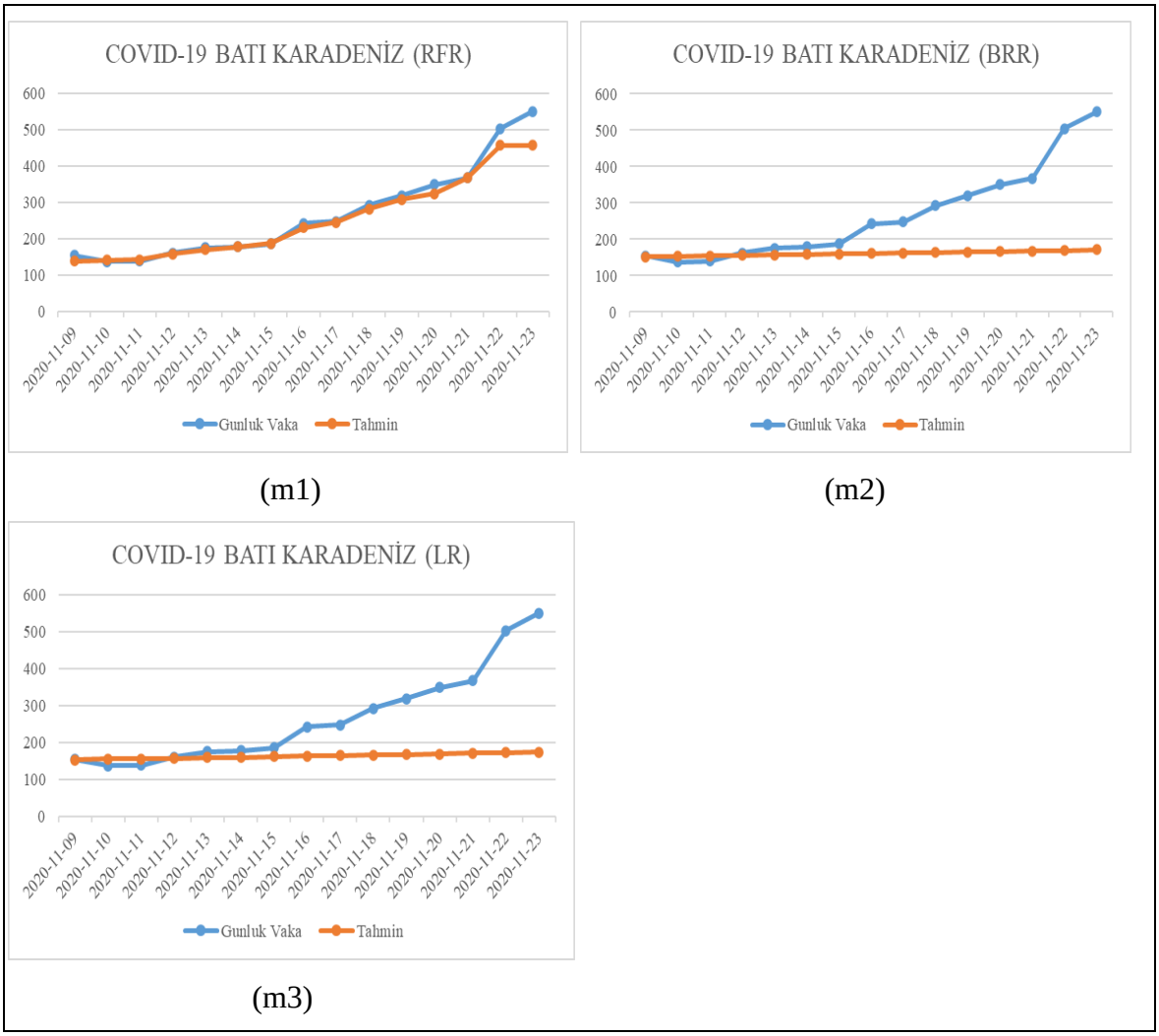


Şekil 4.11. Orta Anadolu k1: RFR; k2: BRR; k3:LR için gerçek ve tahmin değerleri

Çizelge 4.15. Orta Anadolu için tahmin performans değerleri

Orta Anadolu	R^2	RMSE	MSE	MAE
RFR	0,95	15,41	237,48	11,29
BRR	0,19	60,99	3719,99	52,24
LR	0,67	38,68	1496,31	30,77

Şekil 4.11'de Orta Anadolu bölgesi için vaka sayısı artışı görülmektedir. Grafik incelenecek olursa verinin dalgalı bir artış gösterdiği görülür. Bu azalış ve artışlar tahmin güçlüğüne neden olan etkenlerdendir. Çizelge 4.15 incelenecek olursa RFR algoritmasının bu koşullarda bile R^2=0,95 değeri ile etkili bir tahmin gerçekleştirdiği görülür.

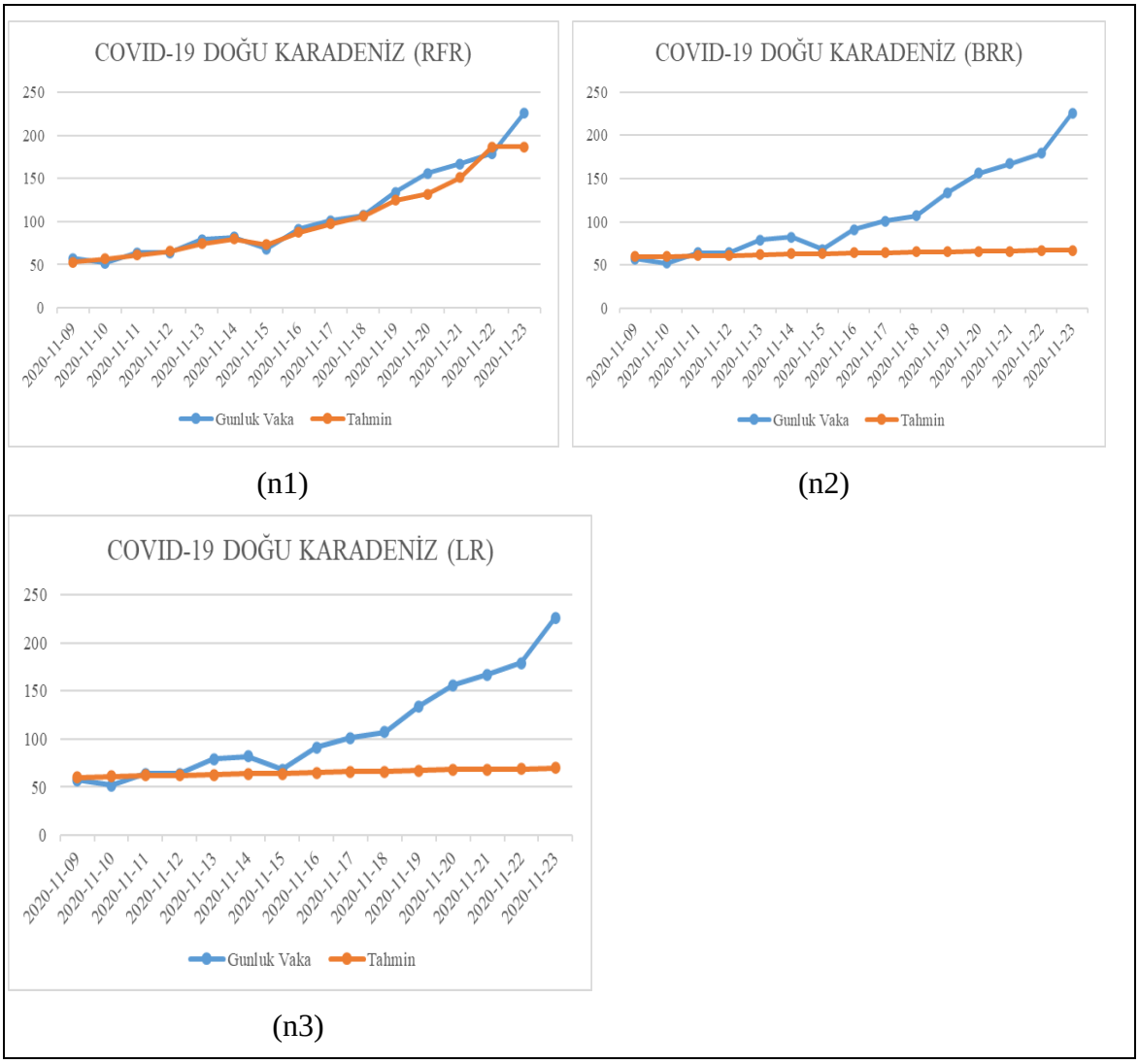


Şekil 4.12. Batı Karadeniz m1: RFR; m2: BRR; m3:LR için gerçek ve tahmin değerleri

Çizelge 4.16. Batı Karadeniz için tahmin performans değerleri

Batı Karadeniz	R^2	RMSE	MSE	MAE
RFR	0,98	12,66	160,38	9,24
BRR	0,42	69,13	4778,91	34,29
LR	0,43	68,79	4732,23	35,06

Şekil 4.12'de Batı Karadeniz için değerler gösterilmektedir. Bu bölge nüfus yoğunluğu düşüktür ancak buna rağmen vaka sayısı artışı hızlıdır. Bölgede tarım, madencilik gibi iş kollarının olması salgının yayılmasına elverişli ortam oluşturmuştur. Çizelge 4.16'e göre bu bölge içinde R^2=0,98 ile RFR en yüksek tahmin performansını göstermiştir.

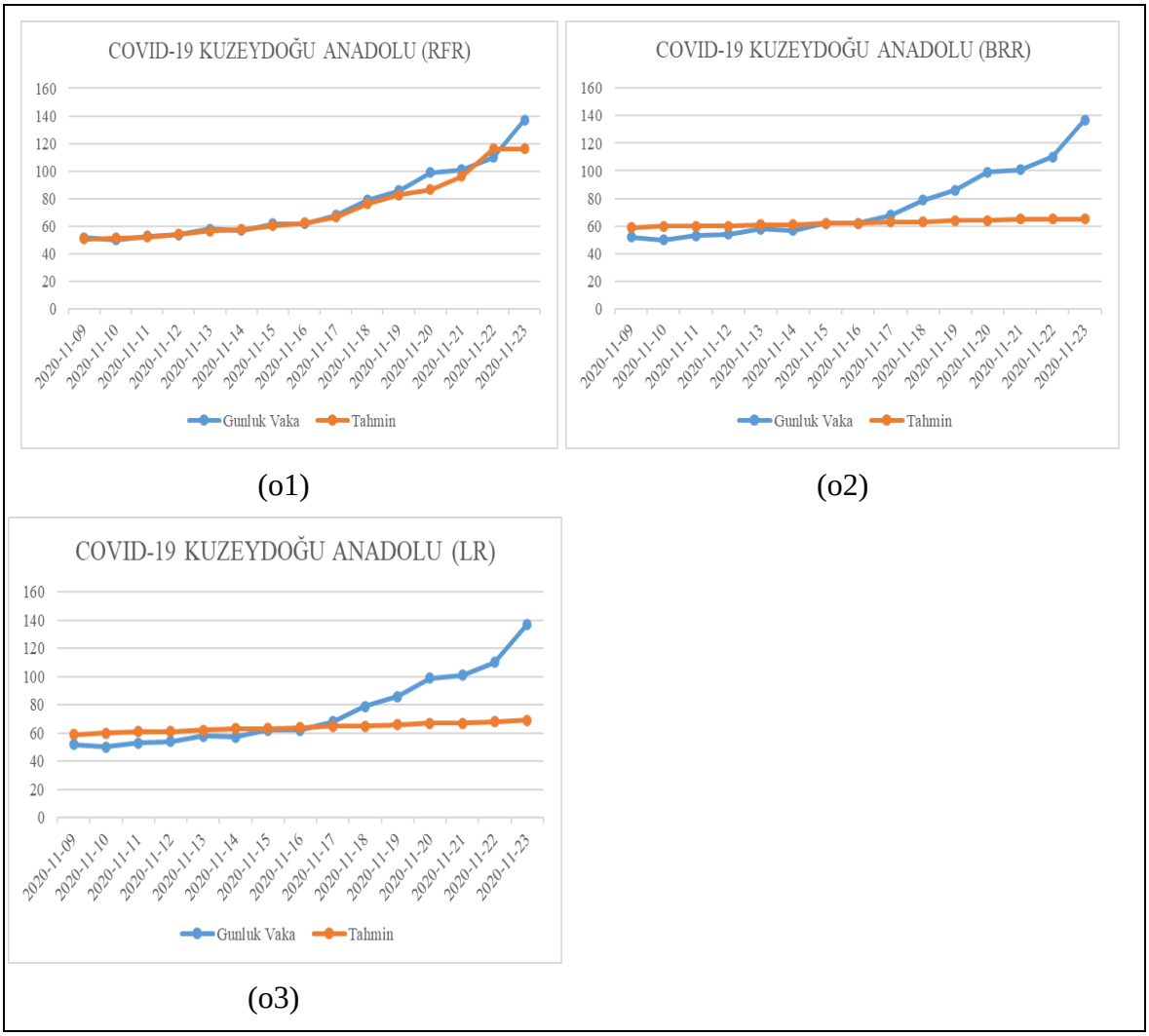


Şekil 4.13. Doğu Karadeniz m1: RFR; m2: BRR; m3:LR için gerçek ve tahmin değerleri

Çizelge 4.17. Doğu Karadeniz için tahmin performans değerleri

Doğu Karadeniz	R^2	RMSE	MSE	MAE
RFR	0,96	6,63	43,9	4,49
BRR	0,33	28,86	832,64	15,52
LR	0,36	28,28	799,86	15,77

Şekil 4.13'te Doğu Karadeniz bölgesi için gerçek ve tahmin değerleri gösterilmektedir. Çizelge 4.17'de BRR yönteminin R^2=0,33 ve LR yönteminin R^2=0,36 olduğu görülmektedir. Bu değerler RFR algoritmasının önemli derecede gerisindedir. RFR yöntemi R^2=0,96 ile gerçeğe çok yakın vaka sayısı tahmininde bulunmuştur.

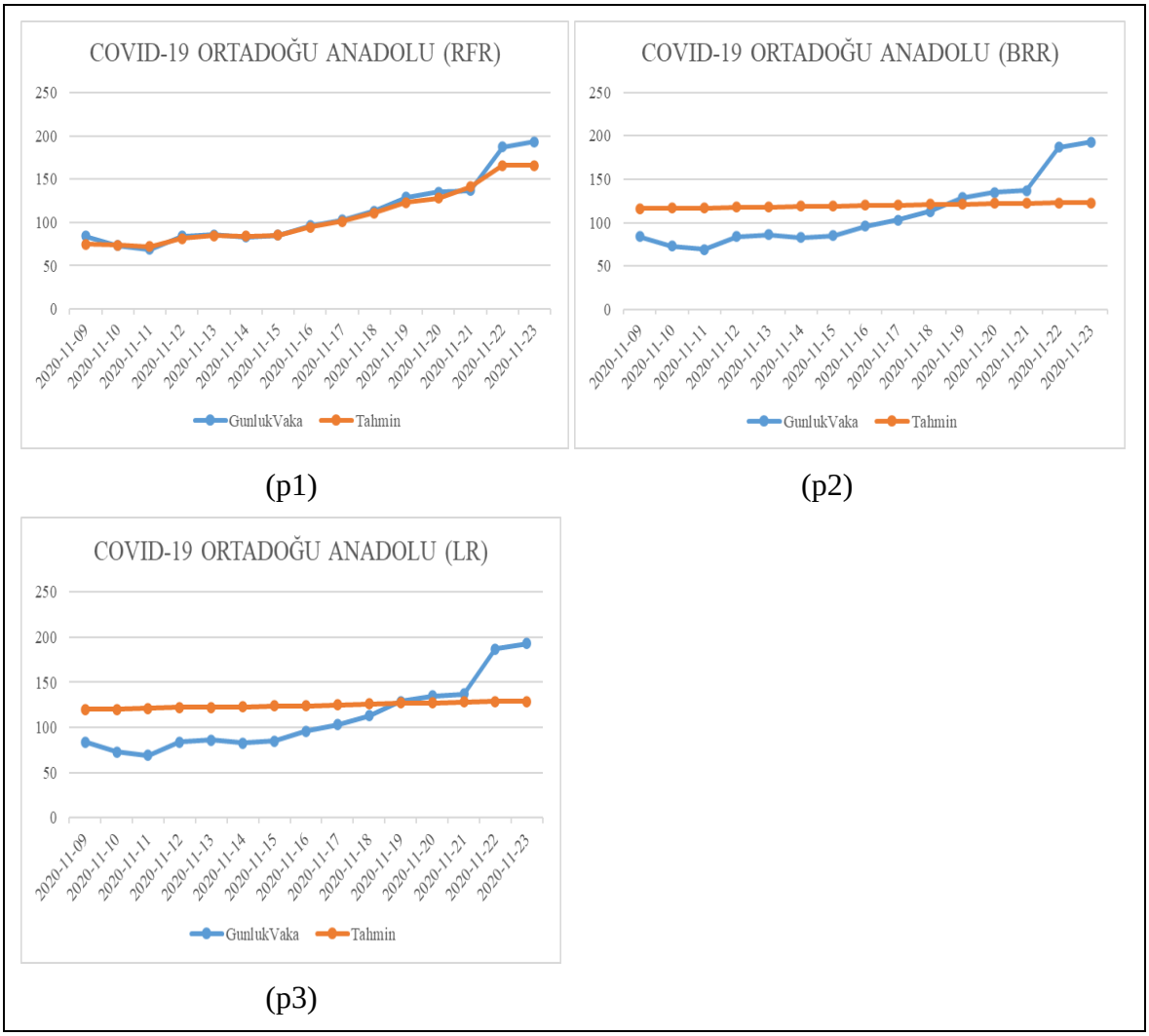


Şekil 4.14. Kuzeydoğu Anadolu o1: RFR; o2: BRR; o3:LR için gerçek ve tahmin değerleri

Çizelge 4.18. Kuzeydoğu Anadolu için tahmin performans değerleri

Kuzeydoğu Anadolu	R^2	RMSE	MSE	MAE
RFR	0,81	9,7	94,04	7,52
BRR	0,17	20,41	416,46	17,09
LR	0,38	17,58	308,94	13,81

Şekil 4.14'te Kuzeydoğu Anadolu için gerçek değerler ile tahmin değerleri gösterilmektedir. Çizelge 4.18'de RFR algoritması R^2=0,81 ile diğer iki yöntemden daha başarılı sonuçlar ortaya koymuştur. RFR yöntemi bu bölgede de RMSE=9,7 değeri ve diğer performans ölçütleri açısından BRR ve LR algoritmasına göre en küçük hata değerine sahiptir.

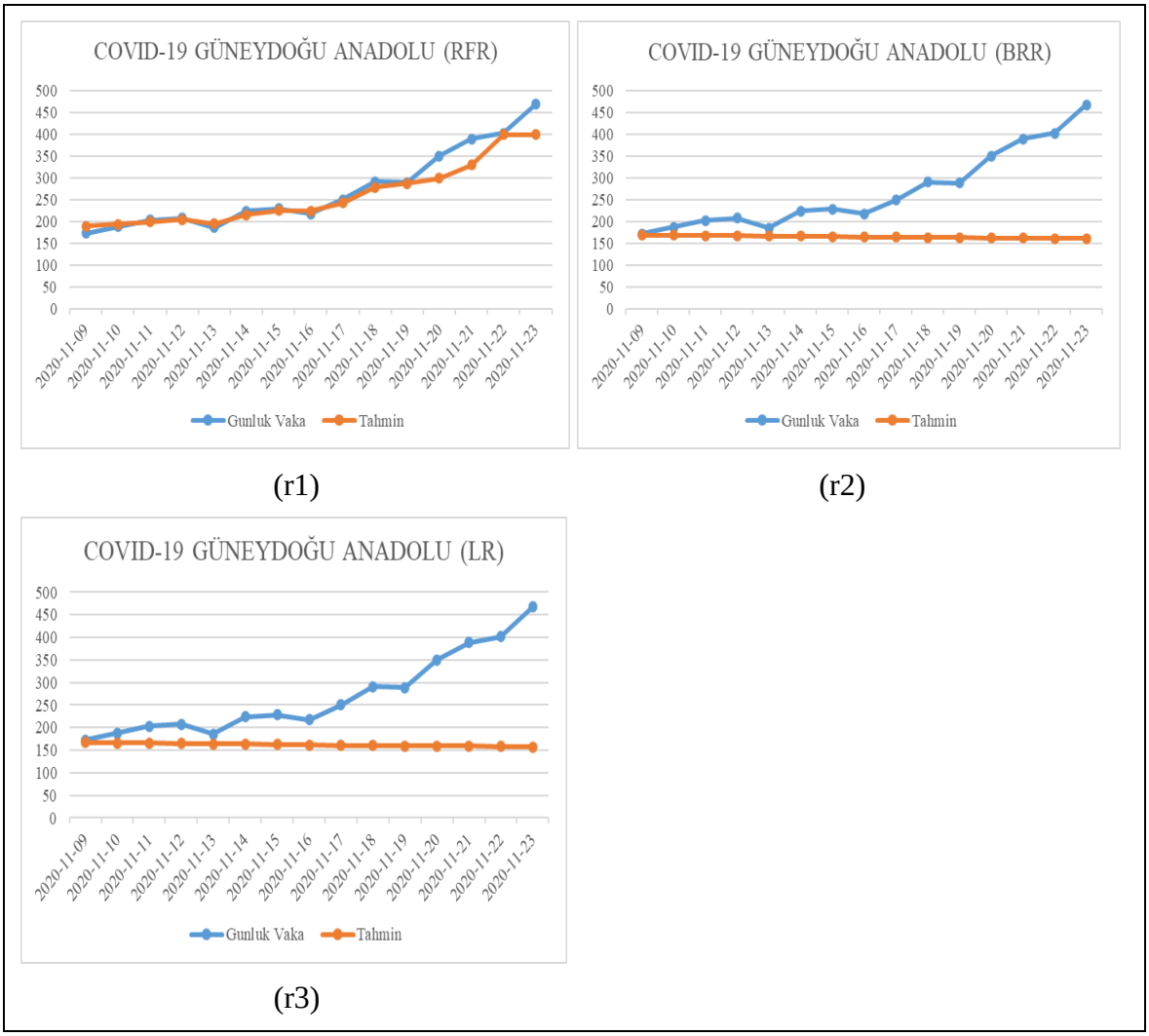


Şekil 4.15. Ortadoğu Anadolu p1: RFR; p2: BRR; p3:LR için gerçek ve tahmin değerleri

Çizelge 4.19. Ortadoğu Anadolu için tahmin performans değerleri

Ortadoğu Anadolu	R^2	RMSE	MSE	MAE
RFR	0,90	13,73	188,38	10,35
BRR	0,31	35,8	1281,33	31,41
LR	0,77	20,64	425,94	16,0

Şekil 4.15'te Ortadoğu Anadolu bölgesinin vaka artış durumu görülmektedir. Çizelge 4.19'da görüldüğü üzere BRR ve LR algoritmaları ile RFR arasında belirgin farklılık vardır. RFR algoritması R^2=0,90 ile gerçeğe yakın bir doğrulukla tahmin yapmayı başarmıştır. RFR'nin RMSE=13,73 değeri de diğer iki yöntemden açık ara daha düşüktür.



Şekil 4.16. Güneydoğu Anadolu r1: RFR; r2: BRR; r3:LR için gerçek ve tahmin değerleri

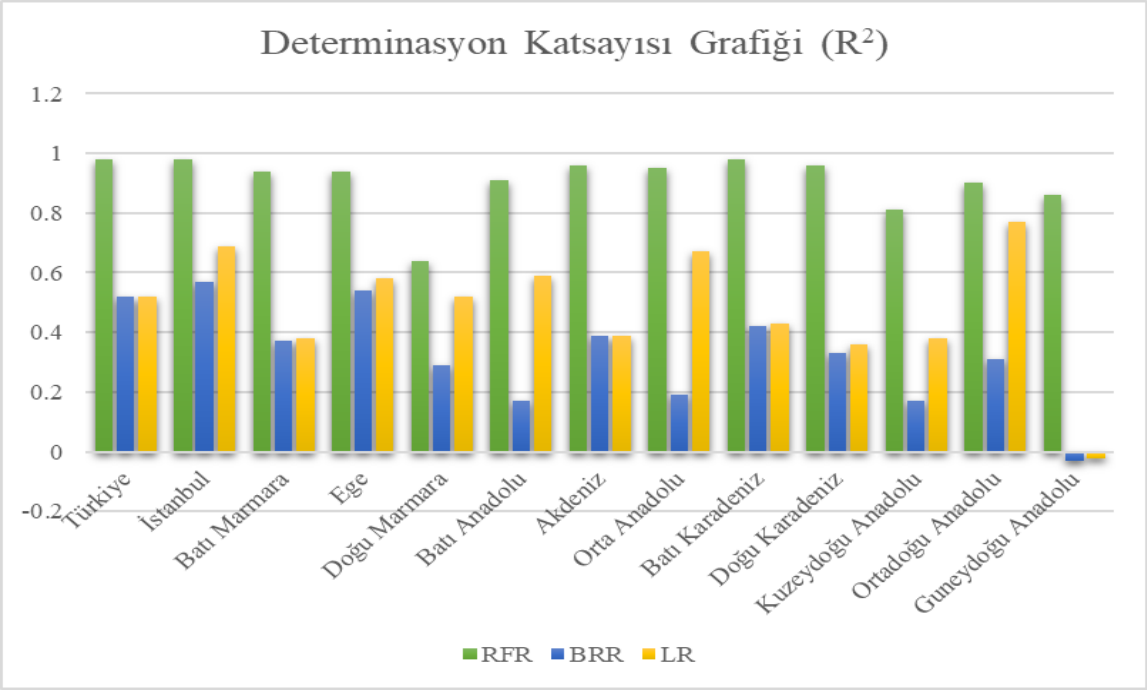
Çizelge 4.20. Güneydoğu Anadolu için tahmin performans değerleri

Güneydoğu Anadolu	R^2	RMSE	MSE	MAE
RFR	0,86	25,17	633,4	20,37
BRR	-0,031	68,12	4640,73	44,22
LR	-0,021	67,8	4596,28	42,03

Şekil 4.16'da Güneydoğu Anadolu için gerçek değerler ile tahmin değerleri gösterilmektedir. Çizelge 4.20'de RFR algoritması R^2=0,86 ile diğer iki yöntemden daha başarılı sonuçlar ortaya koymuştur. BRR ve LR algoritması sırasıyla R^2=-0,031 ve R^2=-0,021 değerleri ile ortalama değerlerden uzak kötü tahmin performansı göstermiştir.

4.5. Tartışma ve Bulgular

Bu çalışmada, Türkiye'nin istatistiki bölgelerinin günlük COVID-19 vaka sayılarının tahmini için regresyon analizleri yapılmıştır. Regresyon analizleri sonucunda elde edilen R^2, RMSE, MSE ve MAE değerleri Çizelge 4.8, Çizelge 4.9, Çizelge 4.10, Çizelge 4.11, Çizelge 4.12, Çizelge 4.13, Çizelge 4.14, Çizelge 4.15, Çizelge 4.16, Çizelge 4.17, Çizelge 4.18, Çizelge 4.19 ve Çizelge 4.20'de ayrı ayrı gösterilmiştir. Modellerin tahmin başarıları, performans değerlendirme metriklerinden R^2 değerine göre belirlenmiştir.



Şekil 4.17. Algoritmaların determinasyon katsayısına göre karşılaştırılması

Makine öğrenimi algoritmalarının determinasyon katsayısına göre karşılaştırılması Şekil 4.17'de görüldüğü üzere Türkiye geneli ve 12 farklı bölge içinde en yüksek R^2 değeri RFR modelinden elde edilmiştir. İstanbul bölgesinde $R^2 = 0{,}98$ değeri ile RFR modeli çok yüksek bir tahmin performansı göstermiştir. Aynı bölge için LR modeli de $R^2 = 0{,}69$ iken BRR modeli $R^2 = 0{,}57$ değeri ile diğer tahmin modellerinin gerisinde kalmıştır. Ayrıca COVID-19 vaka sayısı tahmin performansları her bölge için ayrı ayrı incelendiğinde 12 bölge için RMSE, MSE ve MAE değerlerindeki minimum hatanın yine RFR modeline ait olduğu görülmektedir.

COVID-19 salgını sırasında vaka sayılarında görülen ani artış-azalışlar ve bunların gelecek vakalar üzerine etkisini modellemek ancak belirli periyotlarla yapılan modellemeler ile mümkündür. Bu çalışmada salgının söz konusu değişken süreci makine öğrenimi teknikleri ile analiz edilerek gözle görülür sonuçlar elde edilmiştir.

Türkiye'de COVID-19 salgınının yaygınlığını tahmin etme konusunda yapılan benzer bir çalışmada Toga ve diğerleri (2020), 285 günlük bir örneklem büyüklüğündeki veri setine COVID-19 yaygınlığını tahmin etmek amacıyla otoregresif entegre hareketli ortalama (ARIMA) ve yapay sinir ağı (YSA) tekniklerini uygulamışlardır. Çalışmada ARIMA için R^2= 0,98; YSA için R^2= 0,97 değerleri elde etmişlerdir [79].

Gerçekleştirdiğimiz söz konusu bu çalışmada ise 146 günlük daha küçük bir veri seti ile RFR modeli Türkiye geneli tahmininde R^2 = 0,98 ve ayrıca 12 bölge için elde edilen yüksek R^2 değerleri sonucunda gerçek değerler ile tahminler arasında güçlü ve pozitif bir ilişki olduğunu göstermiştir. Salgınları modellemek hayati derecede önemli olmakla beraber, veriye uygun tahmin yöntemini belirlemek; erken ve doğru tahmin yapmak, salgının vereceği zararı en aza indirgemek için kritik unsurlardandır. YSA'nın tahmin genelleme yeteneği örneklem büyüklüğü ile doğrudan ilişkilidir. YSA genellikle küçük örneklem boyutlarıyla yerel minimumda yakınsar ve zayıf tahmin genellemesi gerçekleştirir [80]. RFR, alt kümelerden rastgele eğitim verisi alarak rastgele ağaçlar oluşturabilmesi nedeniyle diğer makine öğrenmesi algoritmalarından daha güçlüdür [81].

RFR makine öğrenimi yönteminin uygulama kolaylığı ve YSA tabanlı modeller gibi karmaşık modellerden daha verimli olduğu farklı çalışmalarla ortaya konmuş ve gerçekleştirdiğimiz çalışmanın sonuçları ile yeniden kanıtlanmıştır [82].

RFR modelinin uygulandığı farklı alanlardaki sonuçlara bakıldığında da tahmin doğruluğu açısından diğer makine öğrenmesi modellerinden daha iyi performans gösterdiği görülmektedir [26-83-84]. RFR modeli COVID-19 vaka sayısının tahmininde yüksek korelasyon ile birlikte minimum hata değerlerine sahiptir. Bu çalışma, COVID-19 salgınının yayılımını tahmin etmede RFR'nin kabul edilebilir bir doğruluk derecesi ile güçlü bir genelleme yeteneğinde olduğunu açıkça göstermektedir. Deneysel sonuçlar, bu tür tahmine dayalı çalışmaların yerel yönetimleri önceden uyarabildiğini ve yetkililere ilgili önlemleri alması için zaman tanıyabildiğini göstermiştir [29].

6. SONUÇ VE ÖNERİLER

Sağlık sektörü tıbbi müdahaleye ihtiyacı olan insanlara hizmet etmek için tahmin odaklı teknik çalışmalar için önemli potansiyel bir alandır. COVID-19 salgınının sosyo-ekonomik etkisi göz önüne alındığında, bu hastalığın yayılmasını ölçmek her açıdan önem arz etmektedir. COVID-19'un yaygınlığını öngörebilmek amacıyla çeşitli tahmin modelleme tekniklerini kullanmak, halk sağlığı politikalarının oluşturulmasına etkin bir şekilde rehberlik etmek için önemlidir. Özellikle salgın hastalık döneminde binlerce insanın aynı anda sağlık sistemlerinden hizmet alması gerekir. Bununla birlikte, özellikle kaynak sıkıntısı çeken topluluklarda, mevcut sınırlı kaynakların tahsisini optimize etmek için tahminlerin doğru ve kesin olmasını sağlamak gereklidir. Bu nedenle sağlık sistemlerinin donanım ve hizmet kapasitelerini önceden planlayarak müdahalelere hazırlıklı olunması hayati derecede önemlidir.

Birçok durumda salgınların yayılma dinamiğini analiz etmek amacıyla son derece hassas epidemiyolojik yayılma modelleri oluşturulmuştur. Bu çalışmada ise rastgele orman regresyonu (RFR), bayes sırt regresyonu (BRR) ve lineer regresyon (LR) tahmin modelleri ile 1 Temmuz -23 Kasım 2020 tarihleri arasında Türkiye'nin geneli ve istatistiki 12 bölgesine ait COVID-19 vaka sayıları tahmin edilmiştir. RFR, BRR ve LR algoritmaları, literatürde tahmin yöntemleri ile ilgili uygulamalarda etkili performans göstermeleri nedeniyle tercih edilmiştir. Bölge bazlı vaka sayılarının tahmin edilmesindeki esas amaç farklı iklim ve farklı demografi koşullarının salgınların genel yayılımını etkileyen ana unsurlardan olmasıdır. Ülkeler geneli tahmin modellerinde bölgesel yayılım farklılıklarının oluşturabileceği kısıtlar göz ardı edildiğinden, bölge bazlı tahminlerde daha gerçekçi ve net tahmin modelleri söz konusu olacaktır. Çalışmada uygulanan yöntemler; determinasyon katsayısı (R^2), ortalama hata kareleri karekökü (RMSE), hata kareleri ortalaması (MSE), ortalama mutlak hata (MAE) performans ölçütleri ile değerlendirilmiştir.

Sonuçlara göre İstanbul bölgesinde $R^2 = 0,98$ değeri ile RFR modeli çok yüksek bir tahmin performansı göstermiştir. Aynı bölge için LR modeli de $R^2 = 0,69$ iken BRR modeli $R^2 = 0,57$ değeri ile diğer tahmin modellerinin gerisinde kalmıştır.

Elde edilen sonuçlar her üç yöntemin faydalı sonuçlar ürettiğini ancak RFR tekniğinin diğer yöntemlerden çok daha iyi bir tahmin performansına sahip olduğunu göstermiştir.

RFR, BRR ve LR gibi regresyon algoritmaları sadece yeterli doğruluğa sahip olmakla kalmaz, aynı zamanda diğer yöntemlerden daha güvenilirdir. Tüm bu bulgular, gelecekte karşılaşılabilecek salgınlarda epidemiyolojik yayılımın modellenmesi ve tahmini için makine öğrenimi tabanlı regresyon algoritmalarının yüksek performans potansiyeli olduğuna işaret etmektedir.

Makine öğrenimi algoritmaları güvenilir sonuçlar vermesine rağmen, herhangi bir salgının yaygınlığının tahmini yalnızca önceki gözlemlere veya zaman serisi analitik çıkarımlarına bağlı değildir. Sağlık sistemi olanakları, eğitim, insanların farkındalığı, hava durumu, karantina ve sosyal mesafe vb. gibi daha birçok önemli faktör enfeksiyonun büyüklüğünü etkiler. Araştırmacılar gelecekte bu faktörleri de göz önünde bulundurarak daha etkili tahmin modelleri oluşturabilirler.

Makine öğrenimi yöntemleri doğal dil işleme, metin madenciliği, görüntü işleme gibi birçok karmaşık problemlere geleneksel yöntemlere göre daha başarılı çözümler üretmektedir. Bu çalışmanın sonuçları da göz önünde bulundurulursa makine öğrenimi algoritmaları üstel dağılım gösteren veriler veya çeşitli zaman serisi verileri gibi yüksek öngörülme güçlüğü olan dalgalı veri yapılarını etkili şekilde analiz ederek doğru tahmin sonuçları vermektedir. Buradan hareketle gelecek çalışmalarda makine öğrenimi tekniklerinin yüksek değişkenlik ve belirsizliğin hâkim olduğu üretim prosesleri içinde faydalı sonuçlar üretebileceği düşünülmektedir.

KAYNAKLAR

1. Tan, P.N., Steinbach, M., Karpatne, A., and Kumar, V. (2019). *Introduction to Data Mining* (İkinci Baskı). New York: Pearson Education Limited, 21-41.

2. Simeone, O. (2018). *A Brief Introduction To Machine Learning for Engineers*. Boston: Foundations and Trends, 6-10.

3. Chen, N., Zhou, M., Dong, X., Qu, J., Gong, F., Han, Y., Qiu, Y., Wang, J., Liu, Y., Wei, Y., Xia, J., Yu, T., Zhang, X., and Zhang, L. (2020). Epidemiological and Clinical Characteristics of 99 Cases of 2019 Novel Coronavirus Pneumonia in Wuhan, China: A Descriptive Study. *Lancet*, 395(10223), 507-513.

4. Xu, X., Yu, C., Zhang, L., Luo, L., and Liu, J. (2020). Imaging Features of 2019 Novel Coronavirus Pneumonia. *European Journal of Nuclear Medicine and Moleculer Imaging*, 47(5), 1022-1023.

5. Carrasco-Hernandez, R., Jácome, R., López Vidal, Y., and Ponce de León, S. (2017). Are RNA Viruses Candidate Agents for the Next Global Pandemic? A Review. *Ilar Journal*, 58(3), 343-358.

6. İnternet: WHO. (2020). Naming The Coronavirus Disease (COVID-19) And The Virus That Causes It, URL: https://www.who.int/emergencies/diseases/novel-coronavirus-2019/technical-guidance/naming-the-coronavirus-disease-(covid-2019)-and-the-virus-that-causes-it, Son Erişim Tarihi: 30.07.2021

7. İnternet: Hopkins, J. (2021). Coronavirüs Resource Center. URL: https://coronavirus.jhu.edu/map.html, Son Erişim Tarihi: 11.12.2021

8. Krishna, M.V. (2020). Mathematical Modelling on Diffusion and Control of COVID-19. *Infectious Disease Modelling*, 5, 588-597.

9. Ding, Y., and Gao, L. (2020). An Evaluation of COVID-19 in Italy: A Data-Driven Modeling Analysis. *Infectious Disease Modelling*, 5, 495-501.

10. Gnanvi, J.E., Salako, K.V., Kotanmi, G.B., and Glele, K.R. (2021). On the Reliability of Predictions on Covid-19 Dynamics: A Systematic and Critical Review of Modelling Techniques. *Infectious Disease Modelling*, 6, 258-272.

11. Nokes, D.J., and Anderson, R.M. (1988). The Use of Mathematical Models in the Epidemiological Study of Infectious Diseases and in the Design of Mass Immunization Programmes, *Epidemiology and Infection*, 101(1), 1-20.

12. Santosh, K.C. (2020). COVID-19 Prediction Models and Unexploited Data. *Journal of Medical Systems*, 44(9), 170.

13. Mohamadou, Y., Halidou, A., and Kapen, P.T. (2020). A Review of Mathematical Modeling, Artificial Intelligence and Datasets Used ın the Study, Prediction and Management of COVID-19. *Applied Intelligence*, 50(11), 3913-3925.

14. Jubin James, T.K., Mahzaib, F., and Johri, P. (2021). Covid-19 Future Predictions Using Machine Learning Algorithms. *Turkish Journal of Computer and Mathematics Education*, 12.

15. Aslan, I.H., Demir, M., Wise, M.M., and Lenhart, S. (2020). Modeling COVID-19: Forecasting and Analyzing the Dynamics of the Outbreak in Hubei and Turkey. *medRxiv preprint*.

16. Ozdinc, M., Senel, I., Ozturkcan, S., and Akgul, A. (2020). Predicting the Progress of COVID-19: The Case for Turkey. *Turkiye Klinikleri Journal of Medical Sciences*, 40(2),117-9.

17. Arslan, Ş., Özdemir, M.Y., and Uçar, A. (2020). Nowcasting and Forecasting the Spread of COVID-19 and Healthcare Demand in Turkey, A Modelling Study. *Frontiers in Public Health, 15(*8).

18. Musulin, J., Segota, S.B., Stifanic, D., Lorencin, I., Andelic, N., Sustersic, T., Blagojevic, A., Filipovic, N., Cabov, T., and Markova-Car, E. (2021). Application of Artificial Intelligence-Based Regression Methods in the Problem of COVID-19 Spread Prediction: A Systematic Review, *International Journal of Environmental Research and Public Health*, 18(8). 4287.

19. Sujath, R., Chatterjee, J.M., and Hassanien, A.E. (2020). A Machine Learning Forecasting Model for COVID-19 Pandemic in India. *Stochastic Environmental Research and Risk Assessment*, 34(7), 959-972.

20. Ogundokun, R. O., Lukman, A. F., Kibria, G. B. M., Awotunde, J. B., & Aladeitan, B. B. (2020). Predictive Modelling of COVID-19 Confirmed Cases in Nigeria. *Infectious Disease Modelling*, *5*, 543-548.

21. Shah, S., Iftikhar, A., Khan, M.I., Mansoor, M., Mirza, A. F., Bilal, M. (2020). Predicting COVID-19 Infectious Prevalence Using Linear Regression Tool, *Journal of Experimental Biology and Agricultural Sciences , 8, 1-8.*

22. Saqib, M. (2020). Forecasting COVID-19 Outbreak Progression Using Hybrid Polynomial-Bayesian Ridge Regression Model. *Applied Intelligence*, 51(5), 2703-2713.

23. Amar, L.A., Taha, A.A., and Mohamed, M.Y. (2020). Prediction of the Final Size for COVID-19 Epidemic Using Machine Learning: A Case Study of Egypt. *Infectious Disease Modelling*, 5, 622-634.

24. Sengupta, S., Mugde, S., and Sharma, G. (2020). Covid-19 Pandemic Data Analysis and Forecasting Using Machine Learning Algorithms. *medRxiv preprint*.

25. Yeşilkanat., C.M. (2020). Spatio-Temporal Estimation of the Daily Cases of COVID-19 in Worldwide Using Random Forest Machine Learning Algorithm. *Chaos Solitons Fractals*, 140, 110210.

26. Prakash, K.B., Imambi, S.S., Ismail, M., Kumar, T.P., and Pawan, Y. (2020). Analysis, Prediction And Evaluation Of Covid-19 Datasets Using Machine Learning Algorithms. *International Journal of Emerging Trends in Engineering Research*, 8(5). 381–397.

27. Gupta, M., Jain, R., Arora, S., Gupta, A., Javed Awan, M., Chaudhary, G., and Nobanee, H. (2021). AI-Enabled COVID-19 Outbreak Analysis and Prediction: Indian States vs. Union Territories. *Cmc-Computers Materials & Continua*, 67(1), 933-950.

28. Muhammad, L., Islam, M.M., Usman, S.S., and Ayon, S.I. (2020). Predictive Data Mining Models for Novel Coronavirus (COVID-19) Infected Patients' Recovery. *SN Computer Science*, 1(4), 1-7.

29. Pan, Y., Zhang, L., Yan, Z., Lwin, M.O., and Skibniewski, M.J. (2021). Discovering Optimal Strategies for Mitigating COVID-19 Spread Using Machine Learning: Experience from Asia. *Sustainable Cities and Society*, 75, 103254.

30. Vaishnav, P.K., Sharma, S., and Sharma, P. (2021). Analytical Review Analysis for Screening COVID-19 Disease. *International Journal of Modern Research*, 1(1), 22-29.

31. Malki, Z., Atlam, E.S., Hassanien, A.E., Dagnew, G., Elhosseini, M.A., and Gad, I. (2020). Association Between Weather Data and COVID-19 Pandemic Predicting Mortality Rate: Machine Learning Approaches. *Chaos, Solitons & Fractals*, 138, 110137.

32. Parhusip, H.A. (2020). Study on COVID-19 in the World and Indonesia Using Regression Model of SVM, Bayesian Ridge and Gaussian, *Jurnal Ilmiah Sains*, 20(2), 49-57.

33. Ali, M.D. (2020). *Forecasting and Analysis of COVID-19 Pandemic*. MSc Research Project, School of Computing National College of Ireland, Ireland.

34. Khakharia, A., Shah, V., Jain, S., Shah, J., Tiwari, A., Daphal, P., Warang, M., and Mehendale, N. (2021). Outbreak Prediction of COVID-19 for Dense and Populated Countries Using Machine Learning. *Annals of Data Science*, 8(1), 1-19.

35. Taşdelen, B., and Yıldırım, D.D. (2020). Predicting COVID-19 Cases in Turkey with Poisson Regression and the Effect of Preventions on Incidence Rate Ratio Estimation. *Turkiye Klinikleri Journal of Biostatistics*, 12(3), 293-302.

36. Adusumilli, S., Bhatt, D., Wang, H., Devabhaktuni, V., and Bhattacharya, P. (2015). A Novel Hybrid Approach Utilizing Principal Component Regression and Random Forest Regression to Bridge the Period Of GPS Outages. *Neurocomputing*, 166, 185-192.

37. Wirkert, S. J., Kenngott, H., Mayer, B., Mietkowski, P., Wagner, M., Sauer, P., Clancy, N. T., Elson, D. S., and Maier-Hein, L. (2016). Robust Near Real-Time Estimation of Physiological Parameters from Megapixel Multispectral Images With Inverse Monte Carlo and Random Forest Regression. *International Journal of Computer Assisted Radiology and Surgery*, 11(6), 909-917.

38. Ahmad, M. W., Reynolds, J., and Rezgui, Y. (2018). Predictive Modelling for Solar Thermal Energy Systems: A Comparison of Support Vector Regression, Random Forest, Extra Trees and Regression Trees. *Journal Of Cleaner Production*, 203, 810-821.

39. Singh, B., Sihag, P., and Singh, K. (2017). Modelling of Impact of Water Quality on Infiltration Rate of Soil by Random Forest Regression. *Modeling Earth Systems and Environment*, 3(3), 999-1004.

40. Jakariya, M., Alam, M.S., Rahman, M.A., Ahmed, S., Elahi, M.M.L., Khan A.M.S., Saad, S., Tamim, T.M., Ishtiak, T., Sayem, S.M., Ali, M.S., Akter, D. (2020). Assessing Climate-Induced Agricultural Vulnerable Coastal Communities of Bangladesh Using Machine Learning Techniques. *Science of the Total Environment*, 742, 140255.

41. Ozdemir, S. (2016). *Principles of Data Science* (İkinci Baskı). Mumbai: Packt Publishing Ltd, 4-7.

42. İnternet: Fumo, D. (2017). Types of Machine Learning Algorithms You Should Know. URL: https://towardsdatascience.com/types-of-machine-learning-algorithms-you-should-know-953a08248861, Son Erişim Tarihi:18.12.2021

43. Tekin, A.T., Kaya, T., and Çebi, F. (2020*). Click Prediction in Digital Advertisements: A Fuzzy Approach to Model Selection*. International Conference on Intelligent and Fuzzy Systems, İstanbul, Turkey, 213-220.

44. Guha, S., Rastogi, R., and Shim, K. (1998). CURE: An Efficient Clustering Algorithm for Large Databases. *Association for Computing Machinery Sigmod Record*, 27(2), 73-84.

45. Kumar, R., and Verma, R. (2012). Classification Algorithms for Data Mining: A Survey. *International Journal of Innovations in Engineering and Technology*, 1(2), 7-14.

46. Maulud, D., and Abdulazeez, A.M. (2020). A Review on Linear Regression Comprehensive In Machine Learning. *Journal of Applied Science and Technology Trends*, 1(4), 140-147.

47. Yan, X., and Su, X. (2009). *Linear Regression Analysis: Theory and Computing*. Singapore: World Scientific Publishing Co. Pte. Ltd, 9-39.

48. Girginer, N., ve Cankuş, B. (2008). Tramvay Yolcu Memnuniyetinin Lojistik Regresyon Analiziyle Ölçülmesi: Estram Örneği: Yönetim ve Ekonomi. *Celal Bayar Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 15(1), 181-193.

49. Ostertagová, E. (2012). Modelling Using Polynomial Regression. *Procedia Engineering*, 48, 500-506.

50. McDonald, G.C. (2009). Ridge Regression. *Wiley Interdisciplinary Reviews: Computational Statistics*, 1(1), 93-100.

51. Hoerl, A.E., and Kennard, R.W. (1970). Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*, 12(1), 55-67.

52. Darlington, R.B. (1978). Reduced-Variance Regression. *Psychological Bulletin*, 85(6), 1238.

53. Price, B. (1977). Ridge Regression: Application to Nonexperimental Data, *Psychological Bulletin*. 84(4), 759.

54. Kurtuluş, M. (2001). *Ridge Regresyon Üzerine Bir Çalışma*. Yüksek Lisans Tezi, Gazi Üniversitesi Fen Bilimler Enstitüsü, Ankara.

55. Internet: Koptur, M. (2017). Düzenlileştirme (Regularization). URL: https://makineogrenimi.wordpress.com/2017/05/31/duzenlilestirme-regularization/, Son Erişim Tarihi: 26.11.2021

56. Genovese, C.R., Jin, J., Wasserman, L., and Yao, Z. (2012). A Comparison of the Lasso and Marginal Regression. *The Journal of Machine Learning Research*, 13, 2107-2143.

57. Hastie, T., Taylor, J., Tibshirani, R., and Walther, G. (2007). Forward Stagewise Regression and the Monotone Lasso. *Electronic Journal of Statistics*, 1, 1-29.

58. Vapnik, V.N. (1996). *The Nature of Statistical Learning Theory* (İkinci Baskı). New York: Springer, 181-223.

59. Smola, A. J., and Schölkopf, B. (2004). A Tutorial on Support Vector Regression. *Statistics and Computing*, 14(3), 199-222.

60. Açıkkar, M., Akay, M.F., Aktürk, E., and Güleç, M. (2013). *Intelligent Regression Techniques for Non-Exercise Prediction of VO 2 Max*. 2013 21st Signal Processing and Communications Applications Conference (SIU), Cyprus, Turkey, 1-4.

61. Yakut, E., Elmas, B., ve Yavuz, S. (2014). Yapay Sinir Ağlari ve Destek Vektör Makineleri Yöntemleriyle Borsa Endeksi Tahmini. *Süleyman Demirel Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 19(1), 139-157.

62. Acı, M., Avcı, M., ve Acı, Ç. (2017). Destek Vektör Regresyonu Yöntemiyle Karbon Nanotüp Benzetim Süresinin Kısaltılması. *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, 32(3), 901-907.

63. Kecman, V. (2001). *Learning and Soft Computing, Support Vector Machines, Neural Networks, and Fuzzy Logic Models.* Cambridge: The MIT Press, 256-327.

64. Byrtek, M., O'Sullivan, F., Muzi, M., and Spence, A. M. (2005). An Adaptation of Ridge Regression for Improved Estimation of Kinetic Model Parameters from PET Studies. *IEEE Transactions on Nuclear Science*, 52(1), 63-68.

65. Yang, Y., and Yang, Y. (2020). Hybrid Prediction Method for Wind Speed Combining Ensemble Empirical Mode Decomposition and Bayesian Ridge Regression. *IEEE Access: The Multidisciplinary Open Access Journal*, 8, 71206-71218.

66. Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5-32.

67. Alpaydin, E. (2020). *Introduction To Machine Learning* (Second Edition). Cambridge: MIT press, 1-15.

68. Breiman, L. (1996). Bagging Predictors. *Machine Learning*, 24(2), 123-140.

69. Rodriguez-Galiano, V., Sanchez-Castillo, M., Chica-Olmo, M., and Chica-Rivas, M. (2015). Machine Learning Predictive Models for Mineral Prospectivity: An Evaluation of Neural Networks, Random Forest, Regression Trees and Support Vector Machines. *Ore Geology Reviews*, 71, 804-818.

70. Lahouar, A., and Slama, J.B.H. (2017). Hour-Ahead Wind Power Forecast Based on Random Forests. *Renewable Energy*, 109, 529-541.

71. Liaw, A., and Wiener, M. (2002). Classification and Regression by Random Forest. *R News*, 2(3), 18-22.

72. Peters, J., De Baets, B., Verhoest, N.E., Samson, R., Degroeve, S., De Becker, P., and Huybrechts, W. (2007). Random Forests as A Tool for Ecohydrological Distribution Modelling. *Ecological Modelling*, 207(2-4), 304-318.

73. Shi, Q., Abdel-Aty, M., and Lee, J. (2016). A Bayesian Ridge Regression Analysis of Congestion's Impact on Urban Expressway Safety. *Accident Analysis and Prevention*, 88, 124-137.

74. Köksoy, O. (2006). Multiresponse Robust Design: Mean Square Error (MSE) Criterion. *Applied Mathematics and Computation*, 175(2), 1716-1729.

75. Chai, T., and Draxler, R.R. (2014). Root Mean Square Error (RMSE) or Mean Absolute Error (MAE)?–Arguments Against Avoiding RMSE in the Literature. *Geoscientific Model Development*, 7(3), 1247-1250.

76. İnternet: Furkancan, S. (2020). Covid19 In Turkey By Regions. URL: https://www.kaggle.com/greysky/covid19-in-turkey-by-regions, Son Erişim Tarihi: 12.10.2021

77. Taş, B. (2006). AB Uyum Sürecinde Türkiye Için Yeni Bir Bölge Kavramı: İstatistiki Bölge Birimleri Sınıflandırması (İBBS). *Sosyal Bilimler Dergisi, 8(2).*

78. İnternet: Kesen, Y. (2015). Türkiye İstatistikî Bölge Birimleri Sınıflandırması. URL: https://depo.sosyal-bilgiler.com/gorseller/turkiye-istatistiki-bolge-birimleri siniflandirmasi/, Son Erişim Tarihi: 12.10.2021

79. Toga, G., Atalay, B., and Toksari, M.D. (2021). COVID-19 Prevalence Forecasting Using Autoregressive Integrated Moving Average (ARIMA) and Artificial Neural Networks (ANN): Case of Turkey. *Journal of Infection and Public Health*, 14(7), 811-816.

80. Mao, R., Zhu, H., Zhang, L., and Chen, A. (2006). *A New Method to Assist Small Data Set Neural Network Learning*. Sixth International Conference on Intelligent Systems Design and Applications, Aizhi, China, 17-22.

81. Panov, P., and Džeroski, S. (2007). *Combining Bagging and Random Subspaces to Create Better Ensembles*. International Symposium on Intelligent Data Analysis, Slovenia, 118-129.

82. Desai, S., and Ouarda, T.B. (2021). Regional Hydrological Frequency Analysis at Ungauged Sites with Random Forest Regression. *Journal of Hydrology*, 594, 125861.

83. Palmer, D.S., O'Boyle, N.M., Glen, R.C., and Mitchell, J.B. (2007). Random Forest Models to Predict Aqueous Solubility. *Journal of Chemical Information and Modeling*, 47(1), 150-158.

84. Zahedi, P., Parvandeh, S., Asgharpour, A., McLaury, B.S., Shirazi, S.A., and McKinney, B.A. (2018). Random Forest Regression Prediction of Solid Particle Erosion İn Elbows. *Powder Technology*, 338, 983-992.

EKLER

EK-1. Python makine öğrenimi modellerinin veri ön işleme kodları

```
import pandas as pd
import seaborn as sns
import numpy as np
import numpy
import matplotlib.pyplot as plt
from matplotlib import style
from sklearn.model_selection import cross_val_predict
from sklearn import metrics
from sklearn import svm
%matplotlib inline
sns.set(style="white")
sns.set(style="whitegrid", color_codes=True)
sns.set_palette("husl")

#Preprocessing
from sklearn import preprocessing
from sklearn.model_selection import train_test_split

#metrics
from sklearn.metrics import confusion_matrix, accuracy_score, precision_score, recall_score, f1_score, roc_auc_score, roc_curve, make_scorer
#feature selection
from sklearn.model_selection import GridSearchCV, StratifiedKFold
from sklearn import model_selection

#Random Forest
from sklearn.ensemble import RandomForestClassifier
from sklearn import datasets
from sklearn.feature_selection import SelectFromModel
from sklearn.metrics import accuracy_score
from sklearn.ensemble import RandomForestRegressor
```

EK-1.(devam) Python makine öğrenimi modellerinin veri ön işleme kodları

```
#Regression
from sklearn.linear_model import LogisticRegression, BayesianRidge, LinearRegression
from sklearn import metrics
import statsmodels.api as sm

#LDA
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis as LDA
#seed
import random

# Normalizing continuous variables
from sklearn.preprocessing import MinMaxScaler

## Bokeh
from bokeh.plotting import output_notebook, figure, show
from bokeh.models import ColumnDataSource, Div, Select, Button, ColorBar, CustomJS
from bokeh.layouts import row, column, layout
from bokeh.transform import cumsum, linear_cmap
from bokeh.palettes import Blues8, Spectral3
from bokeh.plotting import figure, output_file, show
## Plotly
from plotly.offline import iplot
from plotly import tools
import plotly.graph_objects as go
import plotly.express as px
import plotly.offline as py
import plotly.figure_factory as ff
py.init_notebook_mode(connected=True)
```

EK-1.(devam) Python makine öğrenimi modellerinin veri ön işleme kodlar

```
from matplotlib import dates
import plotly.graph_objects as go

# Time series
from fbprophet import Prophet
import datetime
from datetime import datetime

# Google BigQuery
from googlesearch import search

#import cdist
from scipy.spatial.distance import cdist

#others
from pathlib import Path
import os
from tqdm.notebook import tqdm
from scipy.optimize import minimize
from sklearn.metrics import mean_squared_log_error, mean_squared_error
from sklearn.metrics import r2_score

# Veri Önişleme
train = pd.read_csv("Istanbul_train.csv",parse_dates=['Tarih'])
train.head()
test = pd.read_csv("Istanbul_test.csv",parse_dates=['Tarih'])
test.head()
pop_info = pd.read_csv('Nufus_data.csv')
pop_info.head()

train_rfm = train.copy()
```

EK-1.(devam) Python makine öğrenimi modellerinin veri ön işleme kodları

```
# Add additional variables
#month
train_rfm['month'] = train_rfm['Tarih'].dt.month
#date
train_rfm['dates'] = train_rfm['Tarih'].dt.day
### do the same for test data
test_rfm = test.copy()
# Add additional variables
#month
test_rfm['month'] = test_rfm['Tarih'].dt.month
#date
test_rfm['dates'] = test_rfm['Tarih'].dt.day
train_rfm.tail()

countries_array = train['Bolge'].unique()
train_first_result = pd.DataFrame()

for i in countries_array:
    # get relevant data
    day_first_outbreak = train_rfm.loc[train_rfm['Bolge']==i]
     date_outbreak                                                                    =
day_first_outbreak.loc[day_first_outbreak['GunlukVaka']>0]['Tarih'].min()

    #Calculate days since first outbreak happened
    day_first_outbreak['days_since_first_outbreak']    =    (day_first_outbreak['Tarih']    -
date_outbreak).astype('timedelta64[D]')

#impute the negative days with 0
day_first_outbreak['days_since_first_outbreak'][day_first_outbreak['days_since_first_o
utbreak']<0] = 0
   train_first_result = train_first_result.append(day_first_outbreak,ignore_index=True)
```

EK-1.(devam) Python makine öğrenimi modellerinin veri ön işleme kodları

```
### do the same for test data
test_first_result = pd.DataFrame()
for i in countries_array:

# get relevant data
   day_first_outbreak = test_rfm.loc[test_rfm['Bolge']==i]
   day_first_outbreak_train = train_rfm.loc[train_rfm['Bolge']==i]
   date_outbreak = day_first_outbreak_train.loc[day_first_outbreak_train['GunlukVaka']>0]['Tarih'].min()

#Calculate days since first outbreak happened
   day_first_outbreak['days_since_first_outbreak'] = (day_first_outbreak['Tarih'] - date_outbreak).astype('timedelta64[D]')

#impute the negative days with 0
day_first_outbreak['days_since_first_outbreak'][day_first_outbreak['days_since_first_outbreak']<0] = 0
   test_first_result = test_first_result.append(day_first_outbreak,ignore_index=True)
test_first_result.tail()
#ecoding countries data
#train
labels, values = pd.factorize(train_first_result['Bolge'])
train_first_result['bolge_id'] = labels
train_first_result.head()

#test
labels, values = pd.factorize(test_first_result['Bolge'])
test_first_result['bolge_id'] = labels
test_first_result.head()
```

EK-2. Python makine öğrenimi RFR modelinin kodları

```
random.seed(123)
X = train_first_result[['bolge_id','month','dates','days_since_first_outbreak']]
y_confirm = train_first_result['GunlukVaka']
#Now we find the best parameters to fit in the Random Forest model, we will use it to measure the feature important in the data
X_train_rf, X_test_rf, y_train_rf, y_test_rf = train_test_split(X, y_confirm, test_size=0.3, random_state=0)
#RF model
rf = RandomForestRegressor(n_estimators=100, random_state=(42))
predict_labels = X.columns
rf.fit(X_train_rf, y_train_rf)
y_pred_rf = rf.predict(X_test_rf)
print('Mean Absolute Error:', round(metrics.mean_absolute_error(y_test_rf, y_pred_rf),2))
print('Mean Squared Error:', round(metrics.mean_squared_error(y_test_rf, y_pred_rf),2))
print('Root Mean Squared Error:', round(np.sqrt(metrics.mean_squared_error(y_test_rf, y_pred_rf)),2), "\n")
#r_square
print("r_square: ", r2_score(y_test_rf, y_pred_rf), "\n")
# Print the name and gini importance of each feature
for feature in zip(predict_labels, rf.feature_importances_):
    print(feature)
confirm_case = rf.predict(test_first_result[['bolge_id','month','dates','days_since_first_outbreak']])
forecaseId = pd.DataFrame(test[['TahminId']])
confirm_case = pd.DataFrame(confirm_case)
final_result_rf = pd.concat([forecaseId,confirm_case],axis=1)
final_result_rf.columns = ['TahminId','GunlukVaka']
final_result_rf.head()
final_result_rf.to_csv('submissionIstanbul_rf.csv',index=False)
```

EK-3. Python makine öğrenimi LR modelinin kodları

```
#construct the OLS model
X = train_first_result[['bolge_id','month','dates','days_since_first_outbreak']]
y_confirm = train_first_result['GunlukVaka']

#Now we find the best parameters to fit in the Linear Reg. model, we will use it to measure the feature important in the data
X_train_lr, X_test_lr, y_train_lr, y_test_lr = train_test_split(X, y_confirm, test_size=0.3, random_state=0)

predict_labels = X.columns

# LR MODEL
lr = LinearRegression()

lr.fit(X_train_lr,y_train_lr)

y_pred_lr = lr.predict(X_test_lr) # make the predictions by the model

print('Mean Absolute Error:', round(metrics.mean_absolute_error(y_test_lr, y_pred_lr),2))
print('Mean Squared Error:', round(metrics.mean_squared_error(y_test_lr, y_pred_lr),2))
print('Root Mean Squared Error:', round(np.sqrt(metrics.mean_squared_error(y_test_lr, y_pred_lr)),2))

#r_square
print("r_square: ", r2_score(y_test_lr, y_pred_lr), "\n")
```

EK-3.(devam) Python makine öğrenimi LR modelinin kodları

```
confirm_case= lr.predict (test_first_result [ [ 'bolge_id' ,' month ',' dates ',' days_since_first_outbreak']])


forecaseId = pd.DataFrame(test[['TahminId']])
confirm_case = round(pd.DataFrame(confirm_case),0)

final_result_lin = pd.concat([forecaseId,confirm_case],axis=1)
final_result_lin.columns = ['TahminId', 'GunlukVaka']

final_result_lin.tail()


final_result_lin.to_csv('submissionIstanbul_lr.csv',index=False)
final_result_lin.head()
```

EK-4. Python makine öğrenimi BRR modelinin kodları

```
X = train_first_result[['bolge_id','month','dates','days_since_first_outbreak']]
y_confirm = train_first_result['GunlukVaka']
#Now we find the best parameters to fit in the Bayesian Ridge model, we will use it to measure the feature important in the data
X_train_br, X_test_br, y_train_br, y_test_br = train_test_split(X, y_confirm, test_size=0.3, random_state=0)
predict_labels = X.columns
clf = BayesianRidge(compute_score=True)
clf.fit(X_train_br, y_train_br)
y_pred_br = clf.predict(X_test_br)
print('Mean Absolute Error:', round(metrics.mean_absolute_error(y_test_br, y_pred_br),2))
print('Mean Squared Error:', round(metrics.mean_squared_error(y_test_br, y_pred_br),2))
print('Root Mean Squared Error:', round(np.sqrt(metrics.mean_squared_error(y_test_br, y_pred_br)),2))
#r_square
print("r_square: ", r2_score(y_test_br, y_pred_br), "\n")
## predict on test set
confirm_case=
clf.predict(test_first_result[['bolge_id','month','dates','days_since_first_outbreak']])
forecaseId = pd.DataFrame(test[['TahminId']])
confirm_case = round(pd.DataFrame(confirm_case),0)
final_result_br = pd.concat([forecaseId,confirm_case], axis=1)
final_result_br.columns = ['TahminId','GunlukVaka']
final_result_br.head()
final_result_br.to_csv('submissionIstanbul_br.csv', index = False)
```

GAZİ GELECEKTİR...