BAŞKENT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİLİĞİ ANABİLİM DALI
BİLGİSAYAR MÜHENDİSLİĞİ TEZLİ YÜKSEK LİSANS PROGRAMI

OTOMOTİV ENDÜSTRİSİNDE MAKİNE ÖĞRENİMİ TEKNİKLERİ KULLANILARAK ARAÇ FİYATLARININ TAHMİN EDİLMESİNE YÖNELİK KARŞILAŞTIRMALI BİR ÇALIŞMA

HAZIRLAYAN

LADEN AKGÖK

YÜKSEK LİSANS TEZİ

ANKARA - 2022

BAŞKENT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİLİĞİ ANABİLİM DALI
BİLGİSAYAR MÜHENDİSLİĞİ TEZLİ YÜKSEK LİSANS PROGRAMI

OTOMOTİV ENDÜSTRİSİNDE MAKİNE ÖĞRENİMİ TEKNİKLERİ KULLANILARAK ARAÇ FİYATLARININ TAHMİN EDİLMESİNE YÖNELİK KARŞILAŞTIRMALI BİR ÇALIŞMA

HAZIRLAYAN

LADEN AKGÖK

YÜKSEK LİSANS TEZİ

TEZ DANIŞMANI

DR. ÖĞR. ÜYESİ MEHMET DİKMEN

ANKARA - 2022

.

BAŞKENT ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

Bilgisayar Mühendisliği Anabilim Dalı Bilgisayar Mühendisliği Tezli Yüksek Lisans Programı çerçevesinde Laden AKGÖK tarafından hazırlanan bu çalışma, aşağıdaki jüri tarafından Yüksek Lisans Tezi olarak kabul edilmiştir.

Tez Savunma Tarihi: … / … / ……..

Tez Adı: Otomotiv Endüstrisinde Makine Öğrenimi Teknikleri Kullanılarak Araç Fiyatlarının Tahmin Edilmesine Yönelik Karşılaştırmalı Bir Çalışma

Tez Jüri Üyeleri (Unvanı, Adı - Soyadı, Kurumu) **İmza**

Dr. Öğr. Üyesi Mehmet Dikmen, Başkent Üniversitesi (Tez Danışmanı)

Doç. Dr. Mehmet Serdar Güzel, Ankara Üniversitesi

Dr. Öğr. Üyesi Tülin Erçelebi Ayyıldız, Başkent Üniversitesi

ONAY

Prof. Dr. Faruk ELALDI

Fen Bilimleri Enstitüsü Müdürü

Tarih: … / … / ……..

BAŞKENT ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZ ÇALIŞMASI ORİJİNALLİK RAPORU

Tarih: 09/06/2022

Öğrencinin Adı, Soyadı: Laden AKGÖK

Öğrencinin Numarası: 21910519

Anabilim Dalı: Bilgisayar Mühendisliği

Programı: Bilgisayar Mühendisliği

Danışmanın Unvanı/Adı, Soyadı: Dr. Öğr. Üyesi Mehmet DİKMEN

Tez Başlığı: Otomotiv Endüstrisinde Makine Öğrenimi Teknikleri Kullanılarak Araç Fiyatlarının Tahmin Edilmesine Yönelik Karşılaştırmalı Bir Çalışma

Yukarıda başlığı belirtilen Yüksek Lisans/Doktora tez çalışmamın; Giriş, Ana Bölümler ve Sonuç Bölümünden oluşan, toplam 57 sayfalık kısmına ilişkin, 09/06/2022 tarihinde şahsım/tez danışmanım tarafından Turnitin adlı intihal tespit programından aşağıda belirtilen filtrelemeler uygulanarak alınmış olan orijinallik raporuna göre, tezimin benzerlik oranı % 3'tür. Uygulanan filtrelemeler:

1. Kaynakça hariç

2. Alıntılar hariç

3. Beş (5) kelimeden daha az örtüşme içeren metin kısımları hariç

"Başkent Üniversitesi Enstitüleri Tez Çalışması Orijinallik Raporu Alınması ve Kullanılması Usul ve Esaslarını" inceledim ve bu uygulama esaslarında belirtilen azami benzerlik oranlarına tez çalışmamın herhangi bir intihal içermediğini; aksinin tespit edileceği muhtemel durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Öğrenci İmzası:

ONAY

Tarih: 09/06/2022

Dr. Öğr. Üyesi Mehmet Dikmen

TEŞEKKÜR

Tez danışmanlığımı üstlenen, çalışmalarım boyunca verdiği destek ve danışmanlığı için sayın hocam Dr. Öğr. Üyesi Mehmet DİKMEN'e, tez savunmamda yer alarak değerli görüşleri ile tezimin son şeklini almasını sağlayan sayın hocalarım Doç. Dr. Mehmet Serdar GÜZEL'e ve Dr. Öğr. Üyesi Tülin ERÇELEBİ AYYILDIZ'a çok teşekkür ederim.

Tez çalışmam sırasında sağladığı imkânlardan dolayı çalışmakta olduğum şirketim ASELSAN'a ve tez çalışmam boyunca yardımını esirgemeyen değerli arkadaşım Dr. Bilge Süheyla Akkoca GAZİOĞLU'na teşekkür ederim.

Ayrıca eğitimim süresince bana destek olan aileme, manevi desteği ile yanımda olan kardeşim Alican AKGÖK'e teşekkür ederim.

ÖZET

Laden AKGÖK

OTOMOTİV ENDÜSTRİSİNDE MAKİNE ÖĞRENİMİ TEKNİKLERİ KULLANILARAK ARAÇ FİYATLARININ TAHMİN EDİLMESİNE YÖNELİK KARŞILAŞTIRMALI BİR ÇALIŞMA

Başkent Üniversitesi Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

2022

Pazarlamada doğru tahminlerin yapılabilmesi, finansal getirisi daha yüksek sonuçlar almak ve stratejik kararların daha doğru verilebilmesi açısından önemlidir. Sadece uzman görüşü ile yapılan tahminler yanlış veya yetersiz olabilir ve şirketlere büyük maddi zararlar verebilir. Bu çalışmada, popüler makine öğrenmesi teknikleri ile otomotiv endüstrisindeki araç fiyatları tahmin edilerek bu soruna bir çözüm sunulmaktadır. Son yıllarda literatürde başta moda ürünleri, perakende/pazar ürünleri olmak üzere bilgisayar, elektronik ürünler ve çevrimiçi ürün satışlarının fiyat tahmininde makine öğrenmesi teknikleri kullanılmaktadır. Bu çalışmada, Karar Ağacı, Rastgele Orman, Destek Vektörü, ve Yapay Sinir Ağları yöntemlerinin bir otomotiv satış veri seti üzerindeki performansları değerlendirilmiş ve karşılaştırılmıştır. Deneylerde, 13 özellikli (seri numarası, yeni fiyat, model, konum, yıl, gidilen kilometre, yakıt türü, şanzıman, araç sahibi türü, kilometre, motor, güç, koltuk, fiyat) ve 6019 örnek içeren bir satış veri seti kullanılmış ve üç aşamalı bir ön işlem uygulanmıştır. Bu ön işlemenin son aşamasında Sıralı, One Hot, İkili ve Frekans kodlama yöntemleri ile kategorik değerler sayısal verilere dönüştürülmüştür. Tüm analizlerde fiyat tahmin hatasının hesaplanmasında K-Katlamalı Çapraz Doğrulama yöntemi kullanılmıştır. Yapılan deneyler sonucunda kullanılan veri seti üzerinde en iyi sonucu veren kodlama yöntemi ile en iyi tahmin yöntemi karşılaştırmalı olarak ortaya konmuştur. Sonuçlar, bu çalışmayı ilgili uygulamalar için potansiyel bir seçim haline getiren bazı ilginç noktalar sunmuştur.

Anahtar Kelimeler: Araç Fiyat Tahmini, Makine Öğrenimi, Ön İşleme, Çapraz Doğrulama

ABSTRACT

A COMPARATIVE STUDY TO FORECAST VEHICLE PRICES IN AUTOMOTIVE INDUSTRY USING MACHINE LEARNING TECHNIQUES

AKGOK Laden

Başkent University Institute of Science

Department of Computer Engineering

2022

Precise estimations in marketing is important in terms of getting results with higher financial returns and making more accurate strategic decisions. Estimations made only by expert opinion can be incorrect or insufficient and cause great financial damage to companies. In this study, a solution to this problem is presented to forecast vehicle prices in the automotive industry by using popular machine learning techniques. In recent years, machine learning techniques have been used in the literature for price estimation of computer, electronic products and online product sales, mainly fashion products, retail/market products. In this study, performances of Decision Tree, Random Forest, Support Vector, and Artificial Neural Networks regression techniques on an automotive sales dataset are evaluated and compared. In experiments, a sales dataset of 6019 samples with 13 features (serial number, new price, name, location, year, mileage driven, fuel type, transmission, owner type, mileage, engine, power, seats, price) was used, and a three-stage pre-processing was applied. In the last stage of this pre-processing, categorical values were converted into numerical data by Label, One Hot, Binary and Frequency coding techniques. In all analyses, K-Fold Cross Validation method was used in the estimation of price prediction error. As a result of the experiments, the best coding and the best estimation method on this data set were revealed comparatively. The results have presented some interesting points which makes this study a potential choice for relevant applications.

Keywords: Vehicle Price Prediction, Machine Learning, Pre-processing, Cross Validation.

İÇİNDEKİLER

TABLOLAR LİSTESİ

Sayfa

ŞEKİLLER LİSTESİ

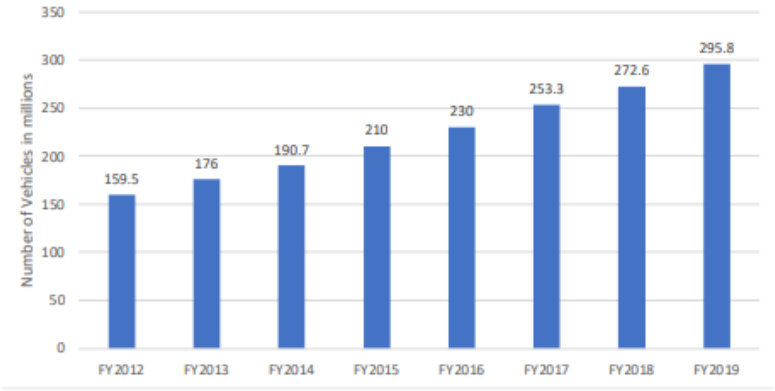
SİMGELER VE KISALTMALAR LİSTESİ

ANN	Artificial Neural Network (Yapay Sinir Ağları)
DT	Decision Tree (Karar Ağacı)
DTR	Decision Tree Regression (Karar Ağacı Regresyonu)
DVM	Destek Vektör Makinesi
kNN	k-Nearest Neighbor (k-En Yakın Komşu)
MAD	Median Absolute Deviation (Ortalama Mutlak Sapma)
MAE	Mean Absolute Error (Ortalama Mutlak Hata)
MAPE	Mean Absolute Percentage Error (Ortalama Mutlak Yüzde Hatası)
MSE	Mean Squared Error (Ortalama Kare Hatası)
RF	Random Forest (Rastgele Orman)
RFR	Random Forest Regression (Rastgele Orman Regresyonu)
RMSE	Root Mean Square Error (Kök Ortalama Kare Hata)
RO	Rastgele Orman
SVM	Support Vector Machine (Destek Vektör Makinesi)
SVR	Support Vector Regression (Destek Vektör Regresyonu)
YSA	Yapay Sinir Ağları

1. GİRİŞ

Tüm dünyada araç talebi göz önüne alındığında, ikinci el otomobil pazarına olan talep artmaktadır. Birçok ülkede, alıcı tarafından uygun fiyatlı olduğundan, kullanılmış bir araba satın almak müşteri için en iyi seçimdir.

Mordor Intelligence [1] tarafından hazırlanan pazar araştırması raporuna göre, ikinci el otomobil pazarı 2021'de 260 milyar ABD Dolar iken tahmin edilen 2022 – 2027 dönemi için %10'un üzerinde bir CAGR (Bileşik Ortalama Büyüme Oranı) ile 2027 yılına kadar 460 milyar ABD Dolarına ulaşması beklenmektedir.



Şekil 1.1. 2012'den 2019'a kadar Hindistan genelinde kayıtlı araç sayısı.

Şekil 1.1'de araç sayısının belirli bir oranla arttığı ve bunun devam ettiği gösterilmektedir. Dolayısıyla, ikinci el otomobil pazarının önemli bir pazar olduğu grafik ile desteklenmektedir [2].

Otomobil fiyatlarının yüksek olması ve otomobile duyulan talebin her geçen gün arttığı düşünüldüğünde ikinci el araçlar alıcı için avantaj haline gelmektedir [3]. Bu alım satımlarda otomobillerin fiyatları belirlenirken, genellikle herhangi bir modellemeden yararlanılmamakta ve benzer ürünlerin fiyatları kıyaslanarak satıcı tarafından bir tutar belirlenmektedir. Bu durum satıcının aracın gerçek değerinden daha yüksek fiyatlarla aracını satmak istemesine ve piyasadaki diğer satıcıların da fiyatları yükseltmesine neden olmaktadır [4].

Yukarıda bahsedildiği üzere otomobil taleplerindeki artış ile otomobil üretimini etkileyen koşullar (COVID-19, çip krizi, vb.) ikinci el araç alımlarında daha yüksek fiyatlardan alıcı bulmak için uygun ortamı oluşturmaktadır [2].

"Limon" pazarı olarak bilinen otomobil pazarında piyasaya göre ortalama kalitede bir araca sahip satıcı, ortalama bir fiyat verilen düşük kaliteli araçların fiyatı ile aynı tutarda aracını satmak istememektedir [5]. Böylece, satıcıyla aynı bilgi düzeyinde olmayan alıcı daha düşük kaliteli araçları değerinden daha yüksek fiyat ile satın almaktadır [2]. Bu durumda kaliteli araçlar, kalitesiz araçlardan daha az satılacaktır, ya da satıcının finansal getirisi daha düşük olacaktır.

Arabaların satış fiyatını doğru tahminleyebilmek basit bir süreç değildir. Bu süreç arabanın özellikleri hakkında uzman bilgisi gerektirmekte olup, sadece uzman görüşü ile yapılacak tahminler yanlış veya yetersiz olabilir. Çünkü aracın fiyatı birçok özelliğe bağlıdır ve özelliklerin listesi sınırlı değildir. Genel olarak otomobilin markası, modeli, yaşı, beygir gücü, gidilen kilometresi, yakıt türü, boyutları, iç özellikleri ve ek paketleri araç fiyatını etkileyecektir [6].

Bu çalışmada alıcı ve satıcı arasındaki fiyat dengesinin sağlanması ve satıcı açısından finansal getirisi daha yüksek sonuçlar alınması için makine öğrenmesi tekniklerinden yararlanılmış ve doğru tahminleme için aracın fiyatını etkileyen 11 özellik (aracın modeli, konumu, yılı, kilometre cinsinden alınan yol, yakıt türü, şanzımanı, araç sahibi türü, kilometresi, motoru, gücü, koltuk sayısı) seçilerek incelenmiştir.

1.1. Araç Fiyat Tahminlemesi Yöntemleri ile İlgili Literatür

Gegic, Isakovic, Keco, Masetic, ve Kevric, [6] araba fiyatı tahmini çalışmasını "web scraper" ile autopijaca.ba web portalından Bosna Hersek'teki kullanılmış arabaların kış mevsimindeki verilerini alarak gerçekleştirmiştir. 1105 örnek içeren ham veri seti toplandıktan sonra, özelliklerin çoğu seyrek ve yararlı bilgi içermediği için kaldırılmıştır. Arabaların renkleri, 15 farklı renk setine normalleştirilmiştir. Yani, veri kayıtları 0-1 dar aralığında bir değere dönüştürülerek, modelin öğrenmesine yardımcı olunmuştur. Küme aralıkları belirlenerek, milaj 5, üretim yılı 7, güç 11, ve fiyat ise 15 farklı kategoriye ayrılmıştır. Böylece, regresyon yöntemleri yerine sınıflandırma yöntemleri ile tahminleme

yapılması sağlanmıştır. Eğitim veri setine üç makine öğrenme tekniği bir arada (Yapay Sinir Ağı, Destek Vektör Makinesi ve Rastgele Orman) "ensemble yöntemi " olarak uygulanması hedeflenmiştir. Öncelikle, arabalar üç fiyat kategorisine ayrılmaktadır: ucuz (fiyat < 12 000 BAM), orta (12 000 BAM <= fiyat < 24 000 BAM) ve pahalı (24 000 BAM <= fiyat). Sonrasında, iki ayrı sınıflandırma yöntemi ile (10 kat ile çapraz doğrulama ve %90 oranında verinin bölünmesi) tüm veri kümesine Rastgele Orman algoritması uygulandığında, %90 oranında verinin bölünmesi daha iyi sonuç verdiği gözlemlenmiştir. Sonuç olarak, %90 veri kümesi bölünmesi durumunda, SVM Ucuz ve Pahalı alt kümelerde en yüksek doğruluğu elde ederken, YSA Orta alt kümede daha iyi performans gösterdiği için ilgili veri setlerine bu iki makine öğrenme tekniği bir arada (Yapay Sinir Ağı, ve Destek Vektör Makinesi) "ensemble yöntemi" olarak uygulanmış ve 87.38% model doğruluğu elde edilmiştir.

Listian, [7] yüksek lisans tezi için yazdığı makalesinde, bir leasing şirketinin doğru fiyatı belirleyebilmesi için ikinci el bir arabanın gelecekteki fiyatını tahmin etmesi gerektiğini değerlendirmektedir. Çoklu doğrusal regresyon analizi yaklaşımının, yüksek boyutlu veriler için uygun olmayacağı düşünülerek, Destek Vektör Regresyonuna başvurulmakta ve bu iki modelin RMSE değerleri karşılaştırılmıştır. Veri seti bir Alman otomobil üreticisinin gerçek dünya verilerinden eksik girdi olmaksızın 180 sütun ve 124.386 örnekten oluşmaktadır. Aracın fiyatı, ay cinsinden ömrü, sürüş kilometresi, 1999'dan beri satıldığı günler, önceki sahiplerin sayısı, müşteri grubu, vergisi, işlem türü, araç kullanım türü, model yılı, serisi, rengi, yastığı, opsiyonel ekipmanları veri işlenirken kullanılan özellikleridir. Regresyon analizinde ve SVR'de bağımsız çok terimli değişkenlerin kullanılması, doğrusal olmayan bir etkiye neden olabileceğinden, bu verilerin ikili değişkene "dummy binary variables" dönüştürülmesi gerekli görülmüştür. Sonrasında normalizasyon uygulanmış ve veri %70 eğitim verisi %30 test verisi olarak ikiye bölünmüştür. SVR'da doğru çekirdeği seçmek için, Gaussian, RBF, ve polinom çekirdeği kullanılarak modelin RMSE değerleri karşılaştırılmıştır ve RBF kullanımı daha uygun görülmüştür. Bir sonraki adımda, aramayı hızlandırmak için optimum bir model oluşturabilecek gerekli minimum eğitim seti boyutu bulunmak istenmiştir. Burada öğrenme eğrisinin amacı, veri seti boyutuna göre SVR'nin hassasiyetini anlamaktır. Bu yüzden artan örnek sayılarıyla SVR birkaç kez çalıştırılarak RSME ölçülmüştür. Sonuç olarak, modelin başarısının seçilen parametrelere bağlı olduğu belirtilerek, SVR modelinin performansı doğrusal regresyon analizine göre daha başarılı bulunmuştur.

Karakoç, [8] makalesinde araba alıp satarken, doğru araç fiyatı tahminin zorluğundan bahsetmiş, ve bu problemin çözümü için "sahibinden.com" araba satış web sitesinden alınan veriler üzerinde tasarlanan iki farklı yapay sinir ağını test etmiştir. Web sitesinden C# programlama dili ve MSSQL veri tabanı yönetim sistemi ile erişilen veri seti, HTMLAgilityPack aracılığı ile okunmuş, ve 1067 araç kaydı elde edilmiştir. Dijitalleştirilen veri setine, sonrasında normalizasyon uygulanmıştır. Girdi kategorileri marka, seri, model, yıl, yakıt, vites, km, gövde tipi, motor gücü, motor hacmi, çekiş, renk, garanti, hasar durumu, plaka, nereden, ticaret ve eyalet iken sonuçta fiyat tahmini yapılmaktadır. Tasarlanan ilk YSA modelinde, 30 ve 25 nötrona sahip iki gizli katman bulunmaktadır. Verinin %70'i eğitim, %30'u ise test verisi olarak kullanılmıştır. İkinci YSA üç gizli katman, sırasıyla 30, 15 ve 25 nötron olacak şekilde oluşturulmuştur. Verilerin %60'ı eğitim için, %40'ı test için kullanılmıştır. İlk YSA için deney 45 iterasyonla çalıştırılmış ve 39 iterasyondan sonra hata arttığı için sonlandırılmıştır. Ikinci YSA için ise deney, 20 iterasyonla çalıştırılmış ve 14 iterasyondan sonra hatanın arttığı gözlemlenerek sonlandırılmıştır. Sonuç olarak, katman ve nöron sayısı artsa da eğitim için kullanılan veri miktarı azaldıkça modelin başarısının azaldığı gözlemlenmiştir. Daha iyi sonuçlar elde edebilmek için daha büyük veri setinin gerekliliğine vurgu yapılmıştır.

Gajera, Gondaliya, ve Kavathiya, [9] yaptığı çalışmasında, ihtiyacı olan insanlara ikinci el araç alımında fiyat tahmininde bulunan bir platform sunmayı hedeflemiştir. Fiyat tahminlemesi, doğrusal regresyon, kNN (k-en yakın komşu), Rastgele Orman, XG artırma ve Karar ağacı gibi makine öğrenimi algoritmaları ile gerçekleştirilmiş ve performansları karşılaştırılmaktadır. Kilometre cinsinden alınan yol, beygir gücü, aracın yaşı, yakıt türü, modeli, markası ve vites tipi aracın fiyat tahminlemesi için kullanılmıştır. Modeli eğitmek için 92.386 kayıt içeren bir veri seti kullanılmıştır. Frekans grafiklerine dayanarak, aykırı ve az bulunduğu gözlemlenen Etanol ve CNG olmak üzere iki yakıt türünün silinmesine karar verilmiştir. Ek olarak, 400.000 Euro'nun üzerinde veya 0 Euro'nun altında fiyatı olan araçların da aykırı değerler olduğu incelenmiş ve veri setinden çıkarılmıştır. Ayrıca, ısı haritasında özellikler arasında korelasyon gözlemlenmemiştir. kNN modeli, 1 ile 100 arasındaki komşuluk değerleri ile çalıştırılmış, 6 komşuluk değerinin en iyi sonucu verdiği gözlemlenmiştir. Sonuç olarak, hem RMSE hem de R-kare sonuçlarına bakıldığında, rastgele orman modelinin performansı daha başarılı bulunmuştur.

Chen, Gu, Deng, ve Huang, [2] yazdığı makalede, kullanılmış araba pazarını analiz ederek, pazardaki kullanılmış araba fiyatlarını araştırmayı ve tahminlemeyi hedeflemiştir. Veri setine Doğrusal, Karar Ağacı, Rastgele Orman ve Gradient Boosting Regresyonu olmak üzere dört makine öğrenme algoritması uygulanmıştır. Kullanılan veri seti, Hindistan'da cardehko.com'da satılan kullanılmış arabalardan elde edilmiş 15000'den fazla veri içeren bir veri seti olup, bu çalışma için Kaggle.com adresinden alınmıştır. Ferrari, Maserati, Mercedes-AMG gibi lüks otomobiller çok az veri (10'dan az) içerdiği için, aykırı kabul edilerek, çıkarılmıştır. Tüm sayısal özelliklerin histogramlarına bakılarak, çoğunun çarpık olduğu gözlemlenmiş ve bu durumun modellemede düşük performansa yol açabileceği çıkarımı yapılmıştır. Sonrasında korelasyona bakılarak gidilen km ve koltuklar gibi bazı özelliklerin, satış fiyatı ile neredeyse hiçbir korelasyona sahip olmadığı, bu nedenle modeller oluşturulurken dikkate alınmayacağı belirtilmiştir. Veri kümesindeki bazı değişkenlerin yüksek varyansları olduğundan ve daha ağır bir şekilde ağırlıklandırılabileceğinden, bunun önüne geçmek için standardizasyon kullanılmış, ve sonrasında kukla değişkenler oluşturulmuştur. Çapraz Doğrulama için yaygın olarak kullanılan kat sayısı 10 olarak ayarlanmış ve her bir algoritmanın sonucunu değerlendirmek için Ortalama Karekök Hatası (RMSE), Ortalama Mutlak Yüzde Hatası (MAPE) ve R-kare (R^2) kullanılmıştır. Daha iyi sonuç veren Random Forest ve Gradient Boosting Regresyon Modelleri üzerinde hiperparametre optimizasyonu yapılmıştır.

Noor ve Jan [10]'ın çalışmaları denetimli makine öğrenimi tekniğini kullanarak bir araç fiyat tahmin sistemi sunmaktadır. Kullanılan veriler, bir veya iki ay içinde pakwheels'ten toplanmıştır. Başlangıçta, kullanılmış 2000 araba kaydı alınmıştır. Toplanan veriler, fiyat, motor kapasitesi, renk, reklam tarihi, görüntüleme sayısı, kilometre, hidrolik direksiyon, alaşım jantlar, şanzıman, motor tipi, kayıtlı şehir, şehir, versiyon, model, marka ve model yılı değişken değerlerini içermektedir. En önemli değişkenler bulunarak, diğer tüm önemsiz değişkenler çıkarılmıştır. Veriler üzerinde ön işleme uygulandıktan sonra geriye 1699 kayıt kalmıştır, metinsel veriler kategori kodlarına, yani 1,0'a dönüştürülmüştür ve fiyat tahmini için çoklu doğrusal regresyon tekniği kullanılarak veriler işlenmiş ve R-kare ile performansı değerlendirilmiştir.

Mauritius, [11] araç fiyatlarını tahmin etmek için çoklu doğrusal regresyon, kNN, Bayes ve karar ağaçları algoritması kullanmıştır. Tahminler, L'Express ve Le Defi gibi günlük gazetelerde bulunan <<petites anons>>'tan toplanmıştır. Her araba için marka,

model, silindir hacmi, kilometre, üretim yılı, boya rengi, manuel/otomatik ve fiyat incelenmiştir. Sütunların çoğu seyrek olduğu için kaldırılmışlardır ve başlangıçta, 400'den fazla kayıt toplanmıştır. Daha fazla budamadan sonra, sadece Toyota, Nissan ve Honda bırakılmış ve 10'dan az kayıt bulunan tüm markalar kaldırılmıştır. Silindir hacmi konusunda, bazı otomobiller için bir aralık seçilmiştir. Nihai veri tabanında sadece 97 kayıt kalmıştır. Grafikte, yıl ile fiyat arasındaki ilişkinin tamamen doğrusal olmadığı görülmüş ve doğrusallığı artırmak için bazı aykırı değerler kaldırılmıştır. Sonrasında, elde edilen veri üzerine uygulanan logaritmik regresyonun basit lineer regresyondan daha iyi olduğu gözlemlenmiştir. 1, 3, 5 ve 10 katlı çapraz doğrulama ile kNN yöntemi uygulanarak, MAE değerleri karşılaştırılmıştır. Son olarak, %80 yüzde bölmesi, 10 kat ile çapraz doğrulama ve eğitim veri seti ile veriler bölünerek karar ağaçları ve bayes modelleri denenmiş ve modellerin performansları karşılaştırılmıştır.

Pal, Arora, ve Sundararaman, [12] makalelerinde ikinci el araba satışlarının artmasının, satıcıların gerçekçi olmayan fiyatlar listeleyerek bu senaryodan yararlanmalarına bağlamış ve araç fiyat tahmini sağlayan bir model ihtiyacını, 500 karar ağacı içeren "Rastgele Orman" modeli çözümü ile karşılamıştır. EBay Kleinanzeigen'den alınan Kaggle'ın İkinci El Araba Veritabanı, 40 markada satılan 370.000'den fazla kullanılmış arabanın fiyatlarını ve 20 çeşit özelliklerini içermektedir. Fiyat üzerinde çok az etkisi olan veya olmayan özellikler ile yok değerindeki, geçersiz ve gerçekçi olmayan bütün veriler veri setinden çıkartılmış ve bazı veriler 0,1 tabanlı sayısal değere dönüştürülmüştür. Girdi verileri 70:20:10 bölme oranıyla eğitim, test verileri ve çapraz doğrulamaya ayrılmıştır. Model tahmini doğruluğu R-kare olarak hesaplanmış ve eğitim puanı %95.82, test puanı %83.63 olarak bulunmuştur. Doğrusal regresyonun doğruluğu, eğitim verilerinde bile %75'in altında çıkmış ve rastgele orman regresyonu aşırı uyum sorununu aştığı için çok daha iyi performans gösterdiği sonucuna varılmıştır.

Madhuvanthi.K vd. [13]'nin çalışması satış tahmininin, tüm ticari şirketlerin gelecekteki hedeflerine ulaşmak için planlarını belirlemesi konusunda yardımcı olduğunu belirtmiştir. AHP-Analitik hiyerarşi süreci, insanların karmaşık kararlar almasına yardımcı olduğu için veri endüstrisi için önemli bulunmaktadır. Otomobil veri setinin özniteliklerine, bu özniteliklerin önemine göre bir puan verilmiş ve sonraki süreçte bu ağırlıklandırılmış özniteliklere göre değerlendirme yapılmıştır. Müşteriye hangi arabanın en iyi olarak derecelendirilebileceği hakkında bilgi vermek için araba satış tahmininde, öznitelikleri

sıralara göre derecelendiren ve müşteriye sunan belirli referans noktaları bulunmaktadır, bu da bulanık mantıktır. Otomobilin derecelerini elde etmek için bu referans noktaları ağırlıklı değerlerle ilişkilendirilmiştir. Ağırlıklı olarak genişlik, yükseklik, beygir gücü, şehir, otoyol ve fiyat gibi altı özellik alınmıştır. Bu çalışmalardan araç satış değerlerinin tahmin edilmesinde etkili olan özelliğin fiyat olduğu sonucuna varılmış ve Analitik Hiyerarşi Sürecinin yardımıyla, doğrusal regresyon, rastgele orman gibi çeşitli makine öğrenme algoritmaları ile çok spesifik ve doğru sonuçlar çıkarılmıştır.

1.2. Diğer Sektörlerde Fiyat Tahminlemesi Yöntemleri ile İlgili Literatür

Bu bölümde, otomotiv sektörü dışındaki diğer sektörlerde karşılaşılan fiyat tahminleme problemlerini inceleyen çalışmalar ele alınmıştır.

Chen, Li, ve Sun, [14] çalışmalarında son yıllarda kripto para fiyatlarındaki düşüş ve yükseliş sebebiyle, Bitcoin'in bir yatırım varlığı olarak görüldüğünden ve değişken yapısından dolayı tahminlemenin doğru yapılmasının öneminden bahsetmiştir. Bu yüzden, makine öğrenmesi tekniklerinden yararlanmanın çalışmalara katkı sağlayacağından bahsedilmiştir. Makale özellik boyutlarının ve farklı makine öğrenme tekniklerinin tahminleme performansına etkisini değerlendirmektedir. Medya ve altın spotu özellikleri de değerlendirilerek Bitcoin günlük fiyat tahmini için daha yüksek boyutlu özellikler oluşturulmaktadır. Bitcoin fiyat verilerini yüksek frekanslı 5 dakikalık aralığa göre ve düşük frekanslı günlük aralığa göre sınıflandırarak örnek boyutunun önemi incelenmiştir. Düşük frekanslı veri seti CoinMarketCap'ten elde edilirken, yüksek frekanslı veri seti 17 Temmuz 2017'den 17 Ocak 2018'e kadar Binance kripto para borsasının API'lerinden gerçek zamanlı Web kazıyıcı oluşturularak toplanmıştır. Ayrıca, başka kaynaklardan toplanan diğer özellikler ile nihai olarak 12 özellik analiz için seçilmiştir. Veri setinin %75'i eğitim için ve kalan %25'i test için kullanılmıştır. Sonrasında farklı makine öğrenme teknikleri ile sonuçlar karşılaştırılmaktadır. Daha yüksek boyutlu özelliklere sahip Bitcoin günlük fiyatı için lojistik regresyon (LR) ve lineer diskriminant analizi (LDA), yüksek frekanslı veri seti için ise Rastgele orman (RF), XGBoost (XGB), ikinci dereceden diskriminant analizi (QDA), destek vektör makinesi (SVM) ve uzun kısa süreli bellek (LSTM) makine öğrenme modelleri ile fiyat tahmini yapılmıştır. Sonuçlar yüksek frekanslı Bitcoin fiyat tahmini için daha karmaşık modellerin benimsenmesi gerektiğini göstermiştir.

Adetunji vd. [15] yaptıkları çalışmada, konut fiyat tahmini için Random Forest makine öğrenimi metodu kullanmıştır. Kullanılan UCI Makine öğrenimi deposu Boston konut veri seti 506 giriş ve 14 özellikten oluşmaktadır. Sayısal değerler normalize edilirken, kategorik değerler birer birer kodlanmıştır. Verilerin keşfedilmesi ve ısı haritasının çıkarımı ile en uygun özelliğin seçilmesinden sonraki aşamada Standart Ölçekleyici ile veriler 1 standart sapması ile 0 civarında kümelenecek şekilde ölçeklendirilmiştir. Her ağaçtan gelen tahminler birleştirilerek, çoğunluğa göre sınıflandırma gerçekleştirilmiştir.

Zhou vd. [16] karayolu demiryolu hemzemin geçitlerinin tasarımının daha güvenli olacak şekilde iyileştirilmesi için yapılan tahminlemede kullanılan karar ağacı ve rastgele orman modellerinin performanslarını karşılaştırmıştır. Federal Demiryolu İdaresi'nin sağladığı veriler, 1996'dan 2014'e kadar Kuzey Dakota'daki 354 kaza ve 5.713 geçişi içermektedir. 30 öznitelik içeren veri setinin %90'ı eğitim, %10'u test için kullanılmıştır. Rastgele Orman modelinde bir ağaç için ayrılan her düğümde rastgele seçilen 6 öznitelik kullanılmış ve bu prosedür modeldeki tüm ağaçlar için tekrarlanmıştır.

Zeng, Liu, ve Yu, [17] bina sistemlerinin enerji tüketimi tahminlerinde kullanılan yöntemlerin karşılaştırmalı bir çalışmasını yapmışlardır. Veri ön işleme sürecinde tüm sıfır değerleri ve büyük sapmalar ortadan kaldırıldıktan sonra, Yapay Sinir Ağı (YSA), Gauss Süreç Regresyonu (GPR), Destek Vektör Makinesi (SVM) ve Çok Değişkenli Doğrusal Regresyon (MLR) yöntemleri R-kare, RMSE ve NMBE metriklerine göre karşılaştırılmıştır.

Sonar, JayaMalini, [18] yayınladıkları makalede bir hastanın diyabetik risk düzeyini daha iyi tahmin edebilecek olan modele karar verebilmek için Karar Ağacı, YSA, Naive Bayes ve SVM modellerini karşılaştırmıştır. Veri kümesinde 768 örneğin %75'i eğitim için ve %25'i test için kullanılmıştır. Ham veri önce anlamlı olacak şekilde dönüştürülmüş, sonrasında önemli görülmeyen özellikler veri setinden çıkarılarak modellerin performansı karşılaştırılmıştır.

Loureiro, Miguéis, Silva, [19] ürünlerin fiziksel özellikleri ve uzmanların görüşleri gibi çeşitli değişkenleri dikkate alarak Karar Ağaçları, Rastgele Orman, Destek Vektör Regresyonu, Yapay Sinir Ağları ve Doğrusal Regresyon modellerini karşılaştırmıştır. Moda ürünlerinin yaşam döngüleri çok kısa olsa da, envanter ve satın alma stratejilerinin belirlenmesi bu büyük miktarda verilerin analiz edilmesi ile desteklenmiştir. Bu çalışmada

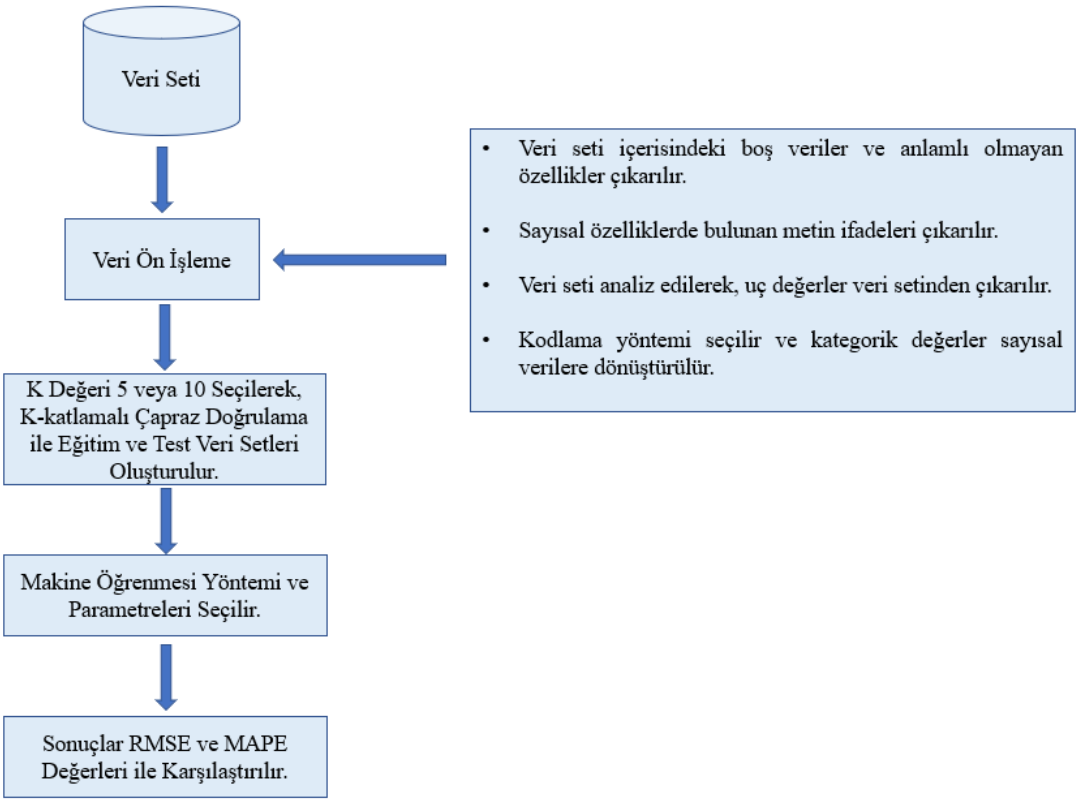
İlkbahar-Yaz 2015 ve İlkbahar-Yaz 2016 sezonlarındaki 684 çeşit bayan çantası verisi kullanılmıştır. 2015 sezonuna ilişkin bilgiler eğitim için 2016 sezonuna karşılık gelen veriler parametre optimizasyonu sonrasında performans değerlendirmesi için test etme amacıyla kullanılmıştır. En iyi parametreler, eğitim veri setinin k katlamalı çapraz doğrulama ile k katlarına bölünmesi ve eğitim-onaylama prosedürünün tanımlanan kat sayı kadar tekrarlanmasıyla bulunmuştur. Sonuçlar R-kare, ortalama karekök hatası (RMSE), ortalama mutlak yüzde hatası (MAPE), ortalama mutlak hata (MAE) ve ortalama kare hatası (MSE) ile değerlendirilmiştir.

Vhatkar, ve Dias, [20] ARIMA, SVM, ANN, Genetik algoritma vb. gibi farklı tahmin tekniklerini kullanarak Ağız Bakım Ürünlerinin satışını tahmin etmiştir. Ürünlerin geçmiş satış verileri toplanarak, en son trendler ve ürünün mevsimsellik değişimleri kontrol edilmiştir. Veri seti aykırı değerler ve eksik verilerden arındırılmıştır. MAD, MSE, RMSE değerleri ile sonuçlar değerlendirilmiştir.

1.3. Problem Tanımı Ve Çalışma Yol Haritası

Bu çalışmada, araç satış fiyatını doğru tahminleyebilmek, alıcı ve satıcı arasındaki fiyat dengesini sağlayabilmek ve satıcı açısından finansal getirisi daha yüksek sonuçlar alınmasını destekleyebilmek için makine öğrenmesi tekniklerinden yararlanılmıştır.

Öncelikle, 13 özellikli (seri numarası, yeni fiyat, model, konum, yıl, gidilen kilometre, yakıt türü, şanzıman, araç sahibi türü, kilometre, motor, güç, koltuk, fiyat) araç veri seti üzerinde veri ön işleme aşaması uygulanmaktadır.



Şekil 1.2 Çalışma Yol Haritası

Veri setindeki aracın seri numarası ve aracın yeni fiyat özellikleri anlamlı olmayan veriler içerdiği için veri setinden çıkarılmıştır. Ayrıca boş veriler içeren araç verileri de veri setinden çıkarılmıştır. Böylece 11 özellikli (model, konum, yıl, gidilen kilometre, yakıt türü, şanzıman, araç sahibi türü, kilometre, motor, güç, koltuk, fiyat) 5872 adet satış verisi elde edilmektedir. Sayısal özelliklerde bulunan metin ifadeleri de çıkarılarak veri seti analiz edilmektedir. Veri setinin normal dağılıma sahip olmasını engelleyen uç değerler veri setinden çıkarılmakta ve son aşamada ise Sıralı Kodlama, One Hot Kodlama, İkili Kodlama ve Frekans Kodlama yöntemlerinden biri seçilerek kategorik değerler sayısal verilere dönüştürülmektedir. K-katlamalı çapraz doğrulama ile eğitim ve test veri setleri oluşturulurken, K 'ya hem 5 hem 10 değeri verilerek, seçilen makine öğrenmesi yöntemi ile RMSE ve MAPE değerlerine göre sonuçlar karşılaştırılmaktadır. Böylece, deneyin sonucunda en düşük hata değerini verecek parametreler ve yöntemler belirlenmiş olacaktır.

2. KULLANILAN YÖNTEMLER

2.1 Kategorik Verilerin Sayısallaştırılması

Bir makine öğrenimi modelinin performansı, yalnızca modele değil, aynı zamanda farklı değişken türlerini modele nasıl işlediğimize de bağlıdır. Son derece değerli olan kategorik veriler orijinal haliyle çoğu model tarafından tanınmadığı için kategorik değişkenlerin sayısal değer olarak kodlanması gerekmektedir. Bu bölümde literatürde kullanılmış olan farklı "kodlama" tekniklerinin model üzerindeki performansı incelenerek, en başarılı kodlama yönteminin belirlenmesi hedeflenmektedir.

Tablo 2.1 Yakıt_Tipi Parametresinin Sıralı Kodlanması.

Yakıt_Tipi	Kod
CNG	0
Diesel	1
LPG	2
Petrol	3

2.1.1 Sıralı Kodlama

Kategorik değerleri kodlamaya yönelik yaklaşımlardan biri olan sıralı kodlaması, bir sütundaki her bir değerin makine tarafından okunabilir olan bir sayıya dönüştürülmesi için kullanılmaktadır. Örneğin, Yakıt_Tipi sütunu 4 farklı değer içerir. Bu sütun Tablo 2.1'deki gibi kodlanmıştır.

2.1.2 One Hot Kodlama

Sıralı kodlama tekniği basit bir yöntem olsa da, sayısal kodlamaları algoritma tarafından yanlış yorumlanabilmektedir. Örneğin, 1 sayısal değeri 3 sayısal değerinden küçük olsa da, hesaplamalarda Diesel, Petrol'dan daha fazla ağırlığı olan bir veri değildir.

Yaygın bir alternatif yaklaşım olan one hot kodlama yönteminde her kategori değeri yeni bir sütuna dönüştürülmekte ve sütuna 1 ya da 0 değeri atanmaktadır. Bu yöntem sayısal değerlerin yanlış şekilde ağırlıklandırılmaması konusunda avantaj sağlarken, veri kümesine daha fazla sütun eklenmesinden dolayı bellek karmaşıklığı konusunda dezavantaja neden

olmaktadır. Bu durumun iyileştirilmesi için kodlamada “get_dummies” özelliği kullanılmaktadır. Bu işlev, kukla/gösterge değişkenleri (1 veya 0) oluşturduğu için bu şekilde adlandırılmıştır.

CNG, Diesel, LPG, Petrol değerlerine sahip olan Yakıt_Tipi sütunu “get_dummies” özelliği kullanılarak bu verilere karşılık gelen 1 veya 0 sayısal değerleri ile dört sütuna dönüştürülmektedir. Bu sütun Tablo 3.2’deki gibi kodlanmıştır.

Tablo 2.2 Yakıt_Tipi Parametresinin One Hot Kodlama Yöntemi ile Kodlanması.

	One Hot Kodlanmış Yakıt_Tipi			
Yakıt_Tipi	**Yt_01**	**Yt_02**	**Yt_03**	**Yt_04**
CNG	1	0	0	0
Diesel	0	1	0	0
LPG	0	0	1	0
Petrol	0	0	0	1

2.1.3 İkili Kodlama

İkili kodlama, sıralı kodlama ile one hot kodlama tekniklerinin bir birleşimidir. Bu teknikte, kategorik özellik ilk önce Sıralı kodlama kullanılarak sayısal veriye dönüştürülür. Daha sonra binary sayıya dönüştürülerek, sayı sütunlara bölünür.

Binary kodlama yönteminde, one hot kodlama yöntemine göre daha az özellik kullanıldığı için, hem bellek açısından daha verimlidir hem de tekrar içermeyen, yüksek boyutlu veriler için boyutluluk problemini azaltır. Çok sayıda kategori içeren verilerin yer aldığı veri setlerinde binary tekniğinin kullanılması one hot kodlamaya göre daha iyi sonuç vermektedir.

CNG, Diesel, Petrol, LPG değerlerine sahip olan body_style sütunu sıralı koda dönüştürülerek, bu verilere karşılık gelen 1 veya 0 sayısal değerleri ile üç sütuna dönüştürülmektedir. Bu sütun Tablo 1.3’teki gibi kodlanmıştır.

Tablo 2.3 Yakıt_Tipi Parametresinin İkili Kodlama ile Kodlanması.

	İkili Kodlanmış Yakıt_Tipi			
Yakıt_Tipi	**Yt_0**	**Yt_1**	**Yt_2**	**Sıralı Kod**
CNG	0	0	1	1
Diesel	0	1	0	2
Petrol	0	1	1	3
LPG	1	0	0	4

2.1.4 Frekans Kodlama

Bir özellik içerisindeki kategoriler (elemanlar) gözlem sayısına göre, 0 ile 1 arasında bir değere dönüştürülmüştür.

Tablo 2.4 Yakıt_Tipi Parametresinin Frekans Kodlanması.

Yakıt_Tipi	**Frekans Kodu**
CNG	0.009644
Diesel	0.527091
Petrol	0.461511
LPG	0.001754

2.2 En İyi Makine Öğrenmesi Yönteminin Seçilmesi

Bu çalışmada kullanılan makine öğrenmesi yöntemleri, yapılan literatür araştırmasına göre kararlaştırılmıştır. Bu çalışmaya göre, son yıllarda fiyat tahminlemesinde en çok Karar Ağacı [2, 9, 11, 18, 19], Rastgele Orman [6, 9, 2, 12, 13, 14, 15, 16, 19], Destek Vektörü [6, 7, 14, 17, 18, 19, 20], ve Yapay Sinir Ağları [6, 8, 17, 18, 19] yöntemleri kullanılmaktadır.

2.2.1 Karar Ağacı Regresyonu

Karar Ağaçları hem sınıflandırma hem de regresyon analizi için kullanılan parametrik olmayan bir öğrenme modelidir [21]. Ayrık bir değişkeni tahmin etmek için sınıflandırma ağaçları, sürekli bir değişkeni tahmin etmek için ise regresyon ağaçları kullanılır [22].

Karar ağaçlarında, kök düğümler, iç düğümler ve yaprak düğümler olmak üzere üç farklı düğüm mevcuttur [23]. Kök düğüm, iç düğümlere bölünen ilk düğümdür. İç düğümler, modelin veri özelliklerini ve karar kurallarını temsil ederken, yaprak düğümler kararın nihai sonucunu temsil eder [21].

Karar Ağacı, bağımsız değişkenleri yinelemeli olarak homojen bölgelere ayıran karar kurallarından oluşan hiyerarşik bir modeldir [24]. DT model, değişkenler arasındaki ilişkilerin modellenmesini, karar kuralının bulunmasını ve kullanıcının bu kuralı yorumlamasını kolaylaştırmaktadır [25]. Ayrıca, girdi faktörlerinin sonucu etkileyen göreli önemi hakkında net bilgiler sağlamaktadır [26]. DT'lerin ana dezavantajı, gürültülü verilerine karşı duyarlı olmalarıdır [27]. Eğitim verilerindeki küçük farklılıklar, ağaç içindeki her seçim noktasında farklı seçimlere neden olabilmektedir. Bu seçimler tüm alt ağaçları etkilediğinden sonuçtaki etkisi önemli düzeyde olabilmektedir [28]. Kullanılan parametreler Tablo 2.5'de belirtilmiştir.

Tablo 2.5 Karar Ağacı Regresyonu (DTR) Parametreleri.

PARAMETRE ADI	**PARAMETRE**	**AÇIKLAMA**
Criterion	'mse'	Bir bölünmenin kalitesini ölçme işlevidir. Ortalama kare hatası "mse" nin amacı, özellik seçim kriteri olarak varyans azaltmaktır. Her terminal düğümünün ortalamasını kullanarak kaybı en aza indirir.
Splitter	"best"	Her düğümde bölmeyi seçmek için kullanılan stratejidir.
max_depth	None	Ağacın maksimum derinliği. Hiçbiri ise, tüm yapraklar saf olana veya tüm yapraklar bir iç düğümü bölmek için gereken minimum örnek sayısından daha azını içerene kadar düğümler genişletilir.
min_samples_split	2	Bir iç düğümü bölmek için gereken minimum örnek sayısı.

min_samples_leaf	1	Bir yaprak düğümde olması gereken minimum numune sayısı. Herhangi bir derinlikteki bir bölünme noktası, yalnızca sol ve sağ dalların her birinde en az "min_samples_leaf" eğitim örneği bırakırsa dikkate alınacaktır. Bu, özellikle regresyonda modeli yumuşatma etkisine sahip olabilir.
min_weight_fraction_leaf	0	Bir yaprak düğümde olması gereken ağırlıklar toplamının (tüm girdi örneklerinin) minimum ağırlıklı kesri. Numune ağırlığı sağlanmadığında numuneler eşit ağırlığa sahiptir.
max_features	None	En iyi bölünmeyi ararken göz önünde bulundurulması gereken özelliklerin sayısı. Bölme araması, "maksimum özellikler"den fazlasını etkin bir şekilde denetlemeyi gerektirse bile, düğüm örneklerinin en az bir geçerli bölümü bulunana kadar durmaz.
random_state	None	Tahmincinin rastgeleliğini kontrol eder. Özellikler, "ayırıcı" "en iyi" olarak seçilse bile, her bölmede her zaman rastgele değiştirilir. "Maksimum özellikler < n özellik" olduğunda, algoritma aralarındaki en iyi bölünmeyi bulmadan önce her bölmede rastgele "maksimum özellikler"i seçecektir. Ancak en iyi bulunan ayrım, "maksimum özellikler=n özellik"' olsa bile farklı çalıştırmalarda değişiklik gösterebilir. Bu durum, kriterdeki iyileştirmenin birkaç bölme için aynı olduğu ve rastgele bir bölmenin seçilmesi gibi bir durumun söz konusu olduğu

		durumdur. Özelliklerin seçimi sırasında deterministik bir davranış elde etmek için "rastgele durum"un bir tamsayıya sabitlenmesi gerekir.
max_leaf_nodes	None	En iyi şekilde "maksimum yaprak düğümleri" olan bir ağaç oluşturmak istenmektedir. En iyi düğümler, saf olmama halinin göreceli azalması olarak tanımlanmaktadır. None seçilmesi, sınırsız sayıda yaprak düğümü anlamına gelmektedir.
min_impurity_decrease	0	Bölünme, bu değerden daha büyük veya buna eşit safsızlıkta bir azalmaya neden olursa, bir düğüm bölünecektir. Ağırlıklı safsızlık azalma denklemi aşağıdaki gibidir: N_t / N * (safsızlık - N_t_R / N_t * sağ_safsızlık - N_t_L / N_t * sol_safsızlık) ``N`` toplam örnek sayısıdır. ``N_t` geçerli düğümdeki örneklerin sayısıdır. ``N_t_L`` sol alt öğedeki örneklerin sayısıdır. ``N_t_R`` sağ alt öğedeki örneklerin sayısıdır. ``örnek ağırlığı`` geçerse, ``N``, ``N_t``, ``N_t_R`` ve ``N_t_L`` nin hepsi ağırlıklı toplamı ifade eder."
min_impurity_split	0	Ağaç büyümesini erken durdurma eşiği. Bir düğüm, safsızlığı eşiğin üzerindeyse bölünür, aksi takdirde bir yapraktır.

		Kaldırılması planlanmaktadır, bunun yerine ``min_impurity_decrease`` kullanılacaktır.
Presort	"deprecated"	Kaldırılacaktır.
ccp_alpha	0	Minimum Maliyetli Karmaşıklık Budaması için kullanılan karmaşıklık parametresi. "ccp alpha"dan küçük, en büyük maliyet karmaşıklığına sahip alt ağaç seçilecektir. Varsayılan olarak, budama yapılmaz.

Ağaçların boyutunu kontrol eden parametreler için varsayılan değerler (örneğin, “max_depth”, “min_samples_leaf”, vb.), bazı veri kümelerinde potansiyel olarak çok büyük olabilen tamamen büyümüş ve budanmamış ağaçlara yol açar. Bellek tüketimini azaltmak için ağaçların karmaşıklığı ve boyutu bu parametre değerleri ayarlanarak kontrol edilir.

2.2.2 Rastgele Orman Regresyonu

Rastgele orman (RF), çok sayıda karar ağacının oluşturulduğu ve bir araya getirildiği bir makine öğrenimi yaklaşımıdır. Herhangi bir düğümün bölünmesi durumunda bir ağaç bir özelliğin alt kümesi olarak oluşturulur ve en iyi özelliği arar, ve böylece n sayıda ağaç oluşturulur [29]. Yeni bir örneğin tahmini ağaçların tahminlerinin ortalaması alınarak bulunur, bu ağaçların oluşturulması ve farklı tahminlerin ortalamasını alma işlemlerine torbalama denir. Ağaç çeşitliliği, rastgele ormanların büyük tahmin gücüne ve ağaçların tahminlerinin ortalamasının alınması azaltılmış varyansa olanak sağlamakta, böylece aşırı öğrenme minimize edilebilmektedir. Ayrıca, ağaçların birbirinden bağımsız olması rastgele ormanın gürültülü verilere karşı dayanıklı olmasını sağlamaktadır [30].

Tablo 2.6 Rastgele Orman Regresyonu (RFR) Parametreleri.

PARAMETRE ADI	PARAMETRE DEĞERİ	AÇIKLAMA
n_estimators	100	Ormandaki ağaç sayısı.
Criterion	Mse	Bir bölünmenin kalitesini ölçme işlevi. Desteklenen kriterler,

		özellik seçim kriteri olarak varyans indirgemesine eşit olan ortalama karesel hata için "mse" ve ortalama mutlak hata için "mae"dir.
max_depth	None	"Ağacın maksimum derinliği. Hiçbiri ise, tüm yapraklar saf olana veya tüm yapraklar min_samples_split örneklerinden daha azını içerene kadar düğümler genişletilir."
min_samples_split	2	"Dahili bir düğümü bölmek için gereken minimum örnek sayısı: "
min_samples_leaf	1	Bir yaprak düğümde olması gereken minimum numune sayısı. Herhangi bir derinlikteki bir bölünme noktası, yalnızca sol ve sağ dalların her birinde en az "min_samples_leaf" eğitim örneği bırakırsa dikkate alınacaktır. Bu, özellikle regresyonda modeli yumuşatma etkisine sahip olabilir.
min_weight_fraction_leaf	0	Bir yaprak düğümde olması gereken ağırlıklar toplamının (tüm girdi örneklerinin) minimum ağırlıklı kesri. Sample_weight sağlanmadığında numuneler eşit ağırlığa sahiptir.
max_features	Auto	En iyi bölünmeyi ararken göz önünde bulundurulması gereken özelliklerin sayısı:

max_leaf_nodes	None	"max_leaf_nodes" ile ağaçları en iyi şekilde büyütün. En iyi düğümler, kirlilikteki göreceli azalma olarak tanımlanır. Hiçbiri ise, sınırsız sayıda yaprak düğümü.
min_impurity_decrease	0	Ağırlıklı safsızlık azaltma denklemi aşağıdaki gibidir: N_t / N * (safsızlık - N_t_R / N_t * right_impurity - N_t_L / N_t * left_impurity) Burada ``N`` toplam örnek sayısıdır, ``N_t`` örnek sayısıdır. geçerli düğümdeki örnekler, ``N_t_L``, düğümdeki örneklerin sayısıdır. sol alt öğe ve ``N_t_R`` sağ alt öğedeki örnek sayısıdır. ``N``, ``N_t``, ``N_t_R`` ve ``N_t_L`` tümü ağırlıklı toplamı ifade eder ("örnek_ağırlık" iletilirse").
min_impurity_split	None	Ağaç büyümesini erken durdurma eşiği. Bir düğüm, safsızlığı eşiğin üzerindeyse bölünür, aksi takdirde bir yapraktır.
Bootstrap	TRUE	Ağaç oluştururken önyükleme örneklerinin kullanılıp kullanılmadığı. False ise, her ağacı oluşturmak için tüm veri kümesi kullanılır.
oob_score	FALSE	Görünmeyen verilerde R^2'yi tahmin etmek için torbadan

		çıkarılmış örneklerin kullanılıp kullanılmayacağını.
n_jobs	None	Paralel olarak çalıştırılacak iş sayısı. :meth:`fit`, :meth:`predict`, :meth:`decision_path` ve :meth:`apply` tümü ağaçlar üzerinde paralelleştirilir. ``Hiçbiri``, bir :obj:`joblib.parallel_backend` bağlamında olmadığı sürece 1 anlamına gelir. ``-1`` tüm işlemcileri kullanmak anlamına gelir.
random_state	None	"Hem ağaç oluştururken kullanılan örneklerin önyüklemesinin rastgeleliğini (eğer ``bootstrap=True`` ise) hem de her düğümde en iyi bölünmeyi ararken dikkate alınacak özelliklerin örneklemesini kontrol eder (eğer ``max_features < n_features` `)."
Verbose	0	Uydurma ve tahmin etme sırasında ayrıntı düzeyini kontrol eder.
warm_start	FALSE	"Doğru" olarak ayarlandığında, topluluğa daha fazla tahmin edici sığdırmak ve eklemek için önceki çağrının çözümünü yeniden kullanın, aksi takdirde tamamen yeni bir ormanı sığdırın.

ccp_alpha	0.0	Minimum Maliyet-Karmaşıklık Budaması için kullanılan karmaşıklık parametresi. "ccp_alpha"dan daha küçük olan en büyük maliyet karmaşıklığına sahip alt ağaç seçilecektir. Varsayılan olarak, budama yapılmaz.

2.2.3 Destek Vektör Regresyonu

Destek Vektör Makineleri (SVM'ler) mevcut biçimleriyle AT&T Bell Laboratuarlarında geliştirilmiş ve kullanımı Cortes ve Vapnik'in makalesi ile ivme kazanmıştır [31]. SVM'ler, girdi vektörlerini yüksek boyutlu (veya sonsuz boyutlu) bir özellik uzayına eşleyerek bir karar yüzeyi oluşturmaktadır. Daha sonra, yüksek boyutlu özellik uzayında doğrusal bir regresyon yürütülmektedir. Bu eşleme işlemine, genellikle çok boyutlu girdi vektörü x ile çıktı y arasındaki ilişki doğrusal olmadığı için ihtiyaç duyulmaktadır. Destek Vektör Makinesi Regresyonu (SVR), çok boyutlu girdi vektörlerini çıktı değerlerine uyduran doğrusal bir hiperdüzlem bulmayı amaçlar. Sonuç, test setindeki bağımlı değişkenin değerlerini tahmin etmek için kullanılmaktadır [32].

Tablo 2.7 Destek Vektör Regresyonu (SVR) Parametreleri.

PARAMETRE ADI	PARAMETRE DEĞERİ	AÇIKLAMA
Kernel	'rbf'	Algoritmada kullanılacak çekirdek türünü belirtir. 'Doğrusal', 'poli', 'rbf', 'sigmoid', 'önceden hesaplanmış' veya çağrılabilirlerden biri olmalıdır.
Degree	3	Polinom çekirdek fonksiyonunun derecesi ('poli').
Gamma	'scale'	rbf', 'poli' ve 'sigmoid' için çekirdek katsayısı.

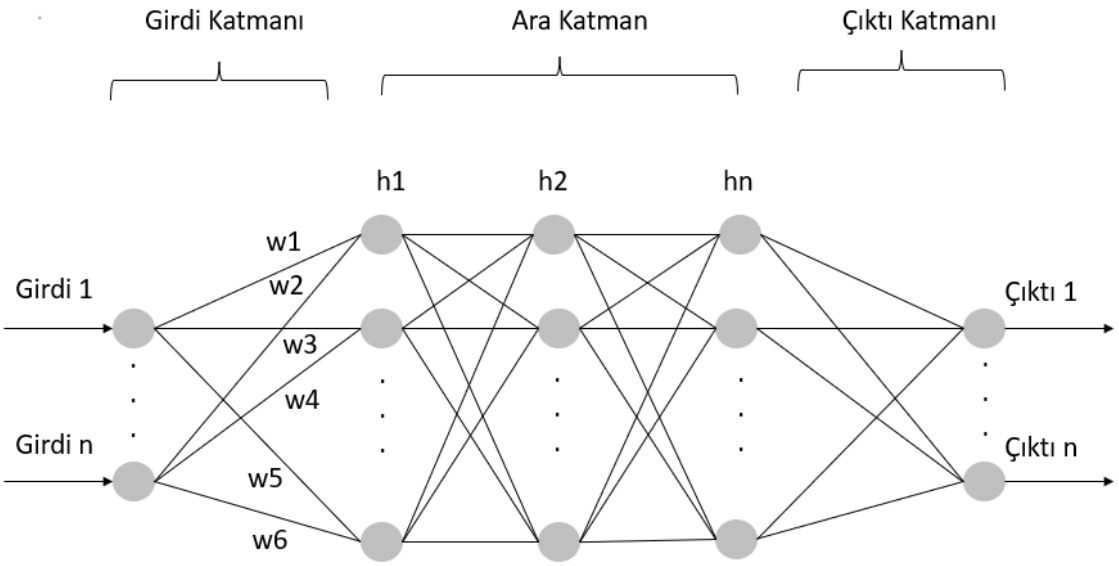
coef0	0.0	Çekirdek işlevinde bağımsız terim. Yalnızca 'poli' ve 'sigmoid'de anlamlıdır.
C	1.0	Düzenlileştirmenin gücü C ile ters orantılıdır. Kesinlikle pozitif olmalıdır.
Epsilon	0.1	Epsilon-SVR modelinde Epsilon. Gerçek değerden bir mesafe epsilon içinde tahmin edilen noktalarla eğitim kaybı fonksiyonunda hiçbir cezanın ilişkilendirilmediği epsilon tüpünü belirtir.
Tol	1e-3'	Durdurma kriteri için tolerans.
Shrinking	TRUE	Shrinking yönteminin kullanılıp kullanılmayacağı.
cache_size	200	Çekirdek önbelleğinin boyutunu belirtin (MB cinsinden).
Verbose	FALSE	libsvm'deki işlem başına çalışma zamanı ayarından yararlanılmaktadır.
max_iter	-1	Problem çözücü içindeki yinelemelerde kesin sınırdır veya sınır yoksa "-1" dir.

2.2.4 Yapay Sinir Ağları

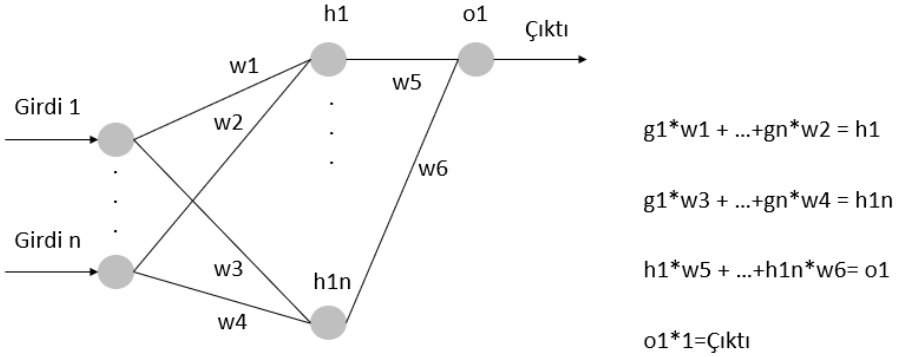
YSA, merkezi sinir sisteminin sinir hücresi (nöronları) ağlarını simüle etmeye çalışan bir hesaplama ağıdır [33]. Herhangi bir YSA, genel olarak Şekil 2.1'deki gibi üç katmana ayrılmış, birbirine bağlı birimler kümesi olarak düşünülebilir. Bu üç katman; girdi katmanı, gizli katman ve çıktı katmanıdır [34]. Sinyaller nöron adı verilen bu bağlı birimler aracılığıyla iletilir. Nöronlar arasındaki bağlantılar, gelen sinyalin ağırlıklandırılması için kullanılır. Net girdi hesaplanırken, farklı fonksiyonlar tercih edilebilir. Genellikle, Şekil 2.1'deki gibi ağırlıklı toplam tercih edilir. Girdiler kendi ağırlığı ile çarpılır ve hepsi toplanır. Her bir nöron için çıkış sinyali, net girişe (ağırlıklandırılmış toplam sinyal) aktivasyonlar uygulanarak elde edilir [35]. Örneğin, aktivasyon fonksiyonu doğrusal fonksiyon olarak seçilirse, net girdi belirlenen bir kat sayı ile çarpılarak çıkış sinyali bulunur. Şekil 2.2'deki örnekte kat sayı 1 olarak belirlenmiştir. Belirli bir işlevi gerçekleştirmek için tasarlanan bir

Yapay Sinir Ağı, nöronlar arasındaki bağlantıların (ağırlıkların) değerlerini belirlemek amacıyla eğitilir [36].

Ayrıca YSA, iyi genelleme yetenekleriyle bilinir ve gürültülü veya eksik verilere karşı büyük ölçüde dayanıklıdır [37]. Gizli katmanların ve her bir gizli katmandaki nöronların sayısı, ağın daha karmaşık fonksiyonlara yaklaşma yeteneği ile orantılıdır. Ancak bu, karmaşık ağ yapılarının her zaman daha iyi performans göstereceği anlamına gelmez [38]. Ağda çok fazla gizli nöron varsa, eğitimsiz veriler için zayıf genellemeye yol açan aşırı parametrelendirme nedeniyle verilerdeki gürültüyü takip edecektir [39]. Öte yandan, çok az gizli nörona sahip bir ağ, gerçek eğilimin yalnızca doğrusal bir tahminine yol açan karmaşık desenler arasında ayrım yapamayacaktır [40].



Şekil 2.1 Üç katmanlıYSA Tasarımı.



Şekil 2.2 Örnek YSA Tasarımı.

2.3 Performans Değerlendirmesinde Kullanılan Metrikler

K-katlamalı çapraz doğrulama ile eğitim ve test veri setleri oluşturulurken, K'ya hem 5 hem 10 değeri verilerek, seçilen makine öğrenmesi yöntemi ile RMSE ve MAPE değerlerine göre sonuçlar karşılaştırılmaktadır. RMSE metriği, hatanın değer olarak daha küçük sayılarla ifade edilmesini sağlar. MAPE metriği hataları yüzde olarak belirtir ve bu şekilde tek başına da bir anlam ifade eder. Bu durum diğer metriklere göre MAPE' yi avantajlı hâle getirir. Performans değerlendirmesinde kullanılan RMSE ve MAPE metriklerinin matematiksel gösterimi aşağıdaki gibidir.

$$RMSE = \frac{1}{n}\sqrt{\sum_{t=1}^{n} \varepsilon_t^2} \tag{2.1}$$

$$MAPE = \frac{1}{n}\sum_{t=1}^{n} \frac{|\varepsilon_t|}{Y_t} \tag{2.2}$$

Y_t = Bir seride t noktasındaki gözlem değeri

$\widehat{Y_t}$ = Yt'nin tahmin değeri

$\varepsilon_t = Y_t - \widehat{Y}_t$ = Hata terimi

n = Gözlem sayısı

Tahmin edilen değer ile gerçekleşen değer arasındaki fark, hata terimi olarak hesaplanmaktadır. Karesi alınarak toplanan hata terimlerinin karekökünün ortalaması ile RMSE metriği elde edilmektedir. MAPE metriğinin hesaplanmasında ise, öncelikle hata terimlerinin mutlak değerleri modelin tahmin ettiği değerlere bölünmektedir. Bulunan bu değerler toplanarak, bu değerlerin ortalaması alındığında elde edilen sonuç MAPE değerini vermektedir.

3. VERİ VE YÖNTEM

3.1 Kullanılan Veri Seti

Bu çalışmada kullanılan 6019 veri içeren 13 özellikli (seri numarası, yeni fiyat, model, konum, yıl, gidilen kilometre, yakıt türü, şanzıman, araç sahibi türü, kilometre, motor, güç, koltuk, fiyat) araç veri seti, Samruddhi ve Kumar [42]'ın çalışmasından alınmıştır. Veri setine Kaggle web sitesinden erişilebilmektedir [43]. Veri setindeki aracın seri numarası ve aracın yeni fiyat özellikleri anlamlı olmayan veriler içerdiği için Samruddhi ve Kumar, [42] 'ın çalışmasındaki gibi veri setinden çıkarılarak, bu çalışmadaki deneylerin ilk aşamasında 11 özellikli (model, konum, yıl, gidilen kilometre, yakıt türü, şanzıman, araç sahibi türü, kilometre, motor, güç, koltuk, fiyat) 5872 adet satış verisi kullanılmıştır. İkinci aşamada ise, hem 5872 adet veri ile hem de veri setinin normal dağılıma sahip olmasını engelleyen uç değerler veri setinden çıkarılarak kalan 5703 veri ile yapılan deneylerin sonuçlarına göre, modellerin performansları karşılaştırılmıştır.

3.2 Verilerin İncelenmesi

Bu çalışmada, öncelikle 11 özellikli girdi verisi olan (model, konum, yıl, gidilen kilometre, yakıt türü, şanzıman, araç sahibi türü, mil, motor, güç, koltuk) ve araba fiyatı olarak 1 çıktısı bulunan 5872 adet satış verisi örneğinin analizi yapılmıştır. Veri üzerinde üç aşamalı bir ön işlem uygulanmıştır. Çıktı parametresi yani "araba fiyatı"nın ortalaması 9.603 olarak hesaplanmıştır. Bu verilerden sayısal değerler içeren özellikler için istatistiksel analizden yola çıkarak Tablo 3.1'deki sonuçlara göre varsayımlarda bulunulabilinir.

Tablo 3.1 Veri Setindeki Sayısal Değerlerin Analizi.

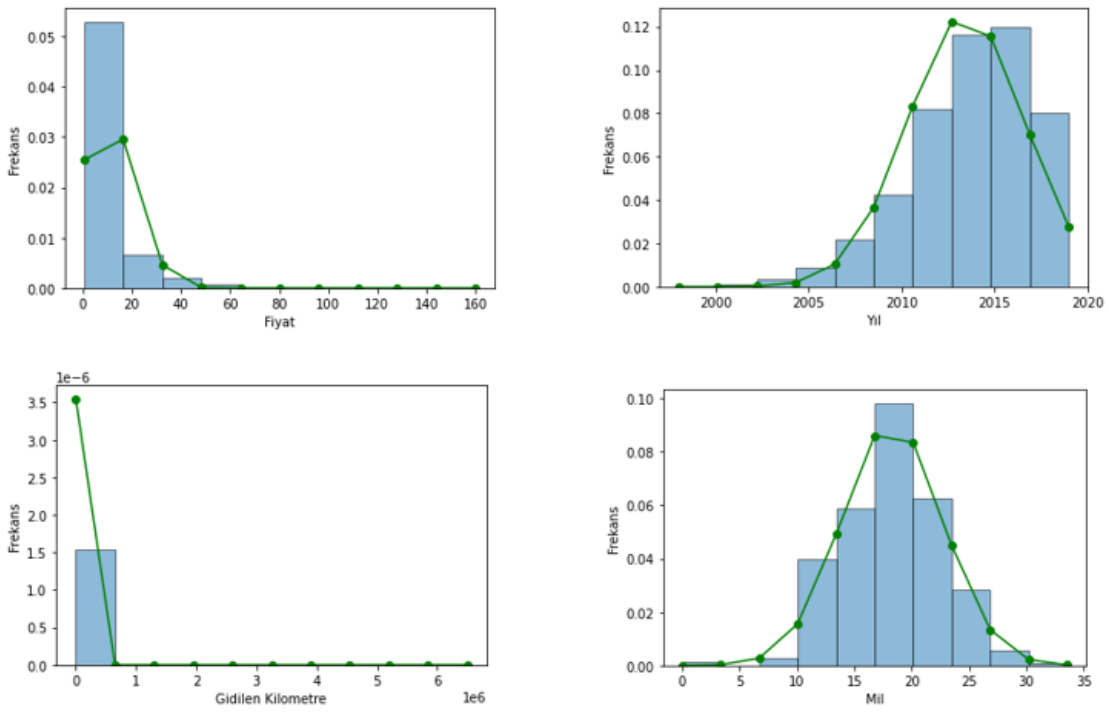
	Fiyat	Yıl	Gidilen Kilometre	Mil	Motor	Güç	Koltuk
count	5872	5872	5872	5872	5872	5872	5872
mean	**9.603**	**2013.477**	**58316.999**	**18.277**	**1625.745**	**113.276**	**5.283**
std	11.249	3.164	92169.410	4.365	601.641	53.881	0.805
min	0.44	1998	171	0	624	34.2	2
25%	3.517	2012	33422.5	15.26	1198	75	5
50%	5.75	**2014**	**52609**	**18.2**	**1495.5**	**97.7**	**5**
75%	**10**	2016	72402.75	21.1	1991	138.1	5
max	160	2019	6500000	33.54	5998	560	10

Genel olarak özelliklerin ortalaması ile medyanı (%50) yaklaşık olarak birbirine yakın gözlemlendiği için ilk aşamada normal dağılıma sahip bir veri seti olduğu söylenebilir.

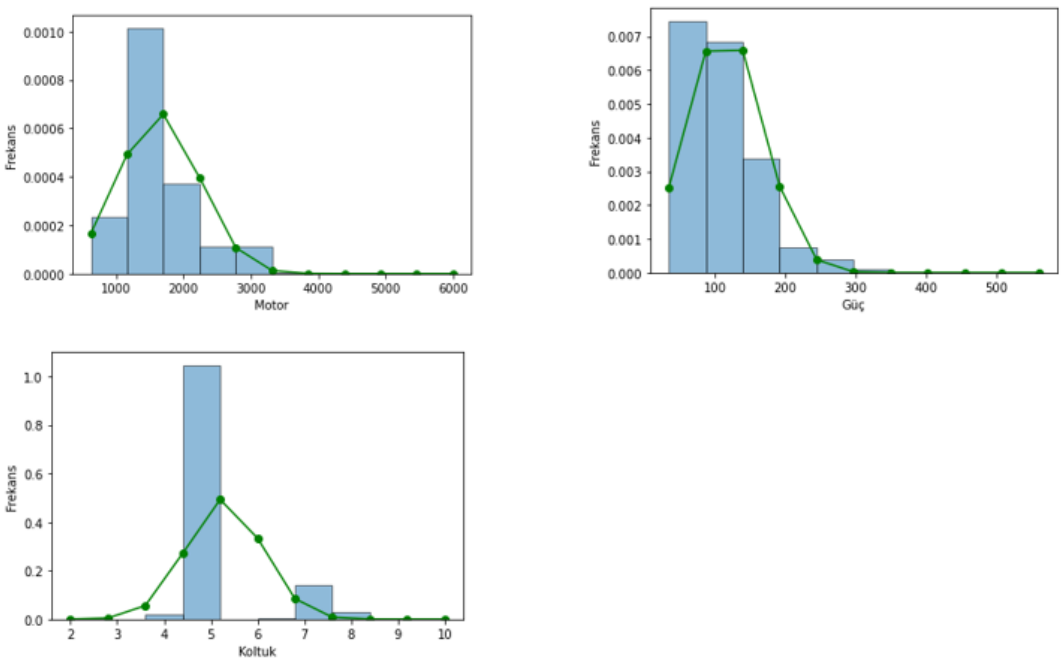
Tablo 3.2 Veri Setindeki Sayısal Değerlerin Dağılımı.

	Fiyat	Yıl	Gidilen Kilometre	Mil	Motor	Güç	Koltuk
mean - 3*std	-24,144	2003,983	-218191,230	5,180	-179,179	-48,368	2,868
Min	0,44	**1998**	171	**0**	624	34,2	**2**
Max	**160**	2019	**6500000**	**33,54**	**5998**	**560**	**10**
mean + 3*std	43,35	2022,97	334825,23	31,37	3430,67	274,92	7,70

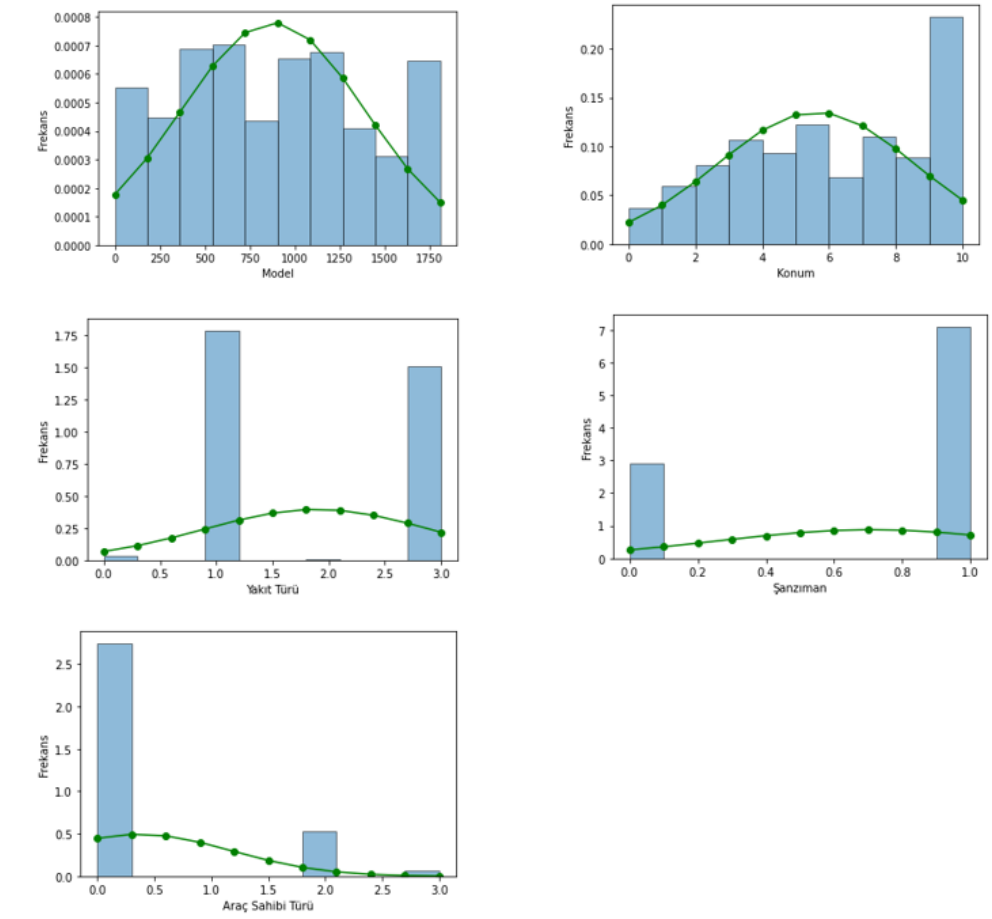
ortalama - 3*std < X < ortalama + 3*std göre veri dağılımın bu değerler arasında yer alması beklenmektedir. Minimum ve maksimum değerler bu aralık içinde yer alıyorsa ayrık değerlerin (outlier) olmadığı varsayılabilir.



Şekil 3.1 Sayısal değerler içeren özelliklerin histogram grafikleri.



Şekil 3.1 devam ediyor.



Şekil 3.2 Sayısal olmayan değerler içeren özelliklerin histogram grafikleri

Veri analizinde incelenen diğer değerler skewness değeri, çarpıklığın yönü hakkında bilgi verirken, kurtosis değeri basıklığın yönü hakkında bilgi vermektedir.

Skewness değeri pozitif ise, veri dağılımı sağa çarpıktır, negatif ise sola çarpıktır.

Kurtosis değerinin 3'ten büyük olması, asimetrik eğrinin mevcut ve uç değerlerin olasılığının yüksek olduğunu gösterirken, 3'ten küçük olması durumu verilerin normal dağılımdan daha basık ve daha geniş bir alana yayılmış olduğunu gösterir, 0'a eşit olması durumu ise simetrik dağılım olduğunu göstermektedir.

Shapiro-Wilk ve Kolmogorov-Smirnov testleri de normal dağılımı test etmek için uygulanabilmektedir. Gözlem sayısı fazla olduğı için Kolmogorov-Smirnov testi de tercih edilmiştir ve sonuçlar aşağıdaki kabule dayalı olarak değerlendirilmektedir:

H0: Değişken normal dağılıma sahiptir. P-Değeri > 0.05
H1: Değişken normal dağılıma sahip değildir. P-Değeri < 0.05

Veri setine Skewness, kurtosis, Shapiro-Wilk ve Kolmogorov-Smirnov testleri uygulanmış ve aşağıdaki sonuçlar elde edilmiştir:

Fiyat:
Skewness: 3.3222398386443843 Kurtosis: 16.94734792656452
Shapiro-Wilk Testi:
T: 0.6415148973464966 P-Değeri: 0.0
Kolmogorov-Smirnov Testi:
T: 0.24143388714418618 P-Değeri: 9.210118472539103e-302

Model:
Skewness: 0.1284212117794548 Kurtosis: -1.056493143194676
Shapiro-Wilk Testi:
T: 0.9609061479568481 P-Değeri: 4.277388901169325e-37
Kolmogorov-Smirnov Testi:
T: 0.06428135720642253 P-Değeri: 1.5440482023444816e-21

Konum:
Skewness: -0.13376366984253193 Kurtosis: -1.1312251093962544
Shapiro-Wilk Testi:
T: 0.943569540977478 P-Değeri: 1.1756894115685215e-42
Kolmogorov-Smirnov Testi:
T: 0.11252001751388052 P-Değeri: 3.2623564820880643e-65

Yıl:
Skewness: -0.8177726429598154 Kurtosis: 0.8952707398145523
Shapiro-Wilk Testi:
T: 0.9524433016777039 P-Değeri: 5.159973309213987e-40
Kolmogorov-Smirnov Testi:
T: 0.1202135633880424 P-Değeri: 2.1065780243771038e-74

Gidilen Kilometre:
Skewness: 58.44156478130156 Kurtosis: 4063.184244711242
Shapiro-Wilk Testi:
T: 0.16816306114196777 P-Değeri: 0.0
Kolmogorov-Smirnov Testi:
T: 0.27059226681187576 P-Değeri: 0.0

Yakıt Türü:
Skewness: 0.15290423805700176 Kurtosis: -1.8966626661625074
Shapiro-Wilk Testi:
T: 0.6579862236976624 P-Değeri: 0.0
Kolmogorov-Smirnov Testi:
T: 0.3593084288568522 P-Değeri: 0.0

Şanzıman:
Skewness: -0.9263974479805122 Kurtosis: -1.1417877683751938
Shapiro-Wilk Testi:
T: 0.5690214037895203 P-Değeri: 0.0
Kolmogorov-Smirnov Testi:
T: 0.44867805747708006 P-Değeri: 0.0

Araç Sahibi Türü:

Skewness: 1.8244926945885585 Kurtosis: 1.62490632624118

Shapiro-Wilk Testi:

T: 0.47967857122421265 P-Değeri: 0.0

Kolmogorov-Smirnov Testi:

T: 0.49986456730991924 P-Değeri: 0.0

Mil:

Skewness: -0.11468888292427755 Kurtosis: 0.7619581807803146

Shapiro-Wilk Testi:

T: 0.9868478775024414 P-Değeri: 5.708153965057273e-23

Kolmogorov-Smirnov Testi:

T: 0.029875082546099607 P-Değeri: 5.4887807747578613e-05

Motor:

Skewness: 1.4165173066293828 Kurtosis: 3.1088909932680986

Shapiro-Wilk Testi:

T: 0.8785017132759094 P-Değeri: 0.0

Kolmogorov-Smirnov Testi:

T: 0.1897315904082708 P-Değeri: 1.4188787988314263e-185

Güç:

Skewness: 1.915716548911364 Kurtosis: 6.501391932513165

Shapiro-Wilk Testi:

T: 0.8431642651557922 P-Değeri: 0.0

Kolmogorov-Smirnov Testi:

T: 0.16101251164518232 P-Değeri: 1.8187479941097463e-133

Koltuk:

Skewness: 1.9145678032215099 Kurtosis: 4.0729445964941915

Shapiro-Wilk Testi:

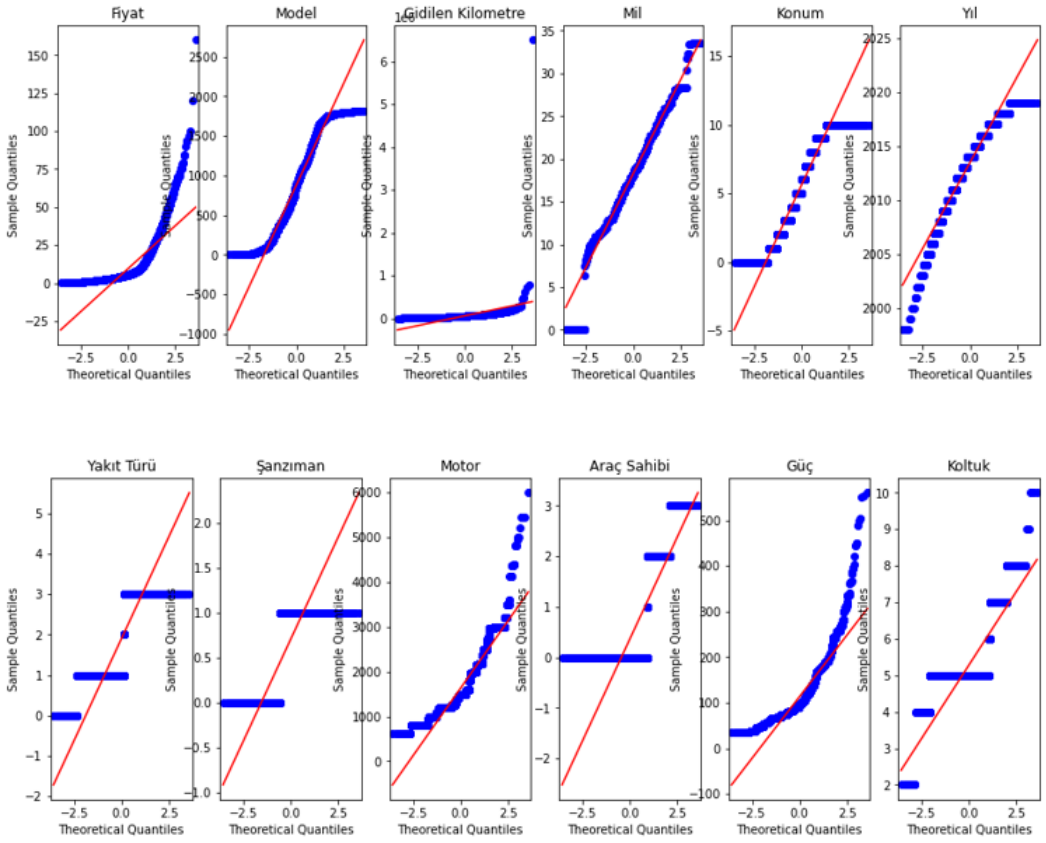
T: 0.5007457733154297 P-Değeri: 0.0

Kolmogorov-Smirnov Testi:

T: 0.49451282291530035 P-Değeri: 0.0

Kolmogorov-Smirnov normallik testi ile veri setindeki fiyat özelliğinin 9.210118472539103e-302 *p* değeri ile %95 güven aralığında ve diğer özelliklerin de 0.05 değerinden az olan p değerleri ile %95 güven aralığında normal dağılım göstermediği gözlemlenmiştir.

QQ (quantile-quantile) grafiğinde, verilerin merkeze yakınlığı normal dağılımın sağlandığı hakkında bilgi vermektedir. Verilerin normal dağılmadığı diğer grafiklerde olduğu gibi Şekil 3.3'te QQ grafiğinde bir çizgide yer almadığı saptanarak gözlemlenmiştir.



Şekil 3.3 Veri setindeki özelliklerin normal dağılımının analizi.

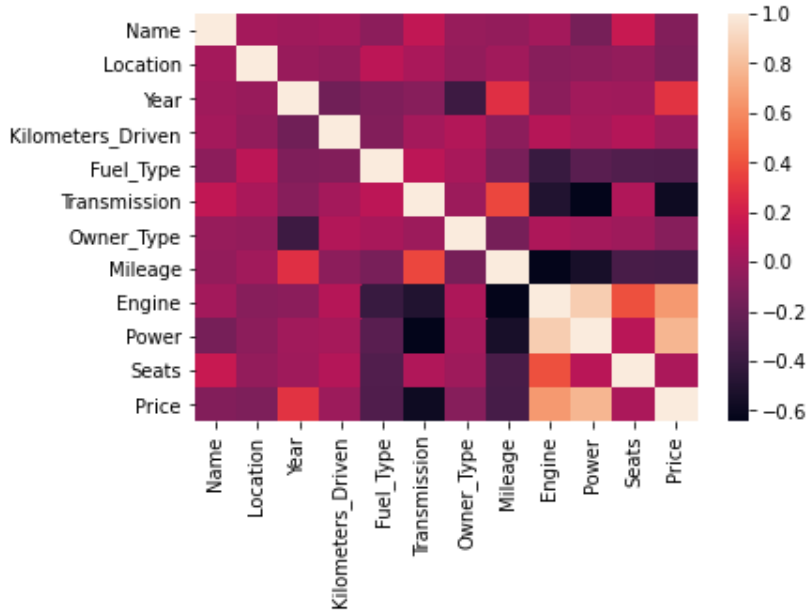
Korelasyon katsayıları aşağıdaki gibi yorumlanmaktadır.
r = -1 → negatif ilişki, r = 1 → pozitif ilişki, r = 0 → ilişki yok

Tablo 3.10'a göre, araba fiyatına en çok etki eden özellikler şanzıman (-0.59), motor (0.66), ve güç (0.77) iken en az etki eden özellikler ise sürülen kilometre (-0.01), koltuk (0.06), araç sahibi türü (-0.09), model (0.10) ve konumdur (-0.12).

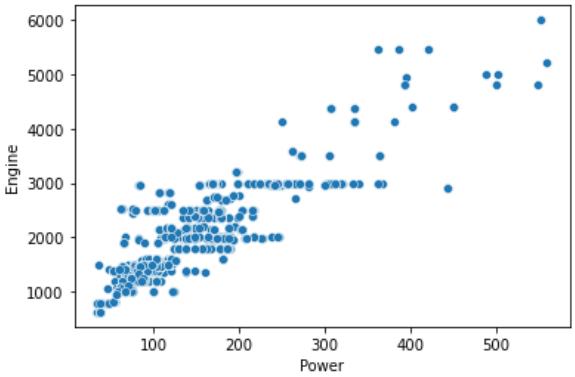
Bunun dışında güç ve motor arasında pozitif yönde (0.87) güçlü bir ilişki gözlemlenirken güç ve şanzıman ile motor ve mil arasında ortalama negatif yönde (-0.64) bir ilişki gözlemlenmektedir. Ayrıca koltuk ve araç sahibi ile şanzıman ve araç sahibi arasında bir ilişki yoktur. Güç ve yıl ile koltuk ve yıl arasındaki ilişki ise yok denecek kadar azdır. Sonuçlar, oluşturulan Şekil 3.4'te ısı haritasında da görsel olarak gösterilmiştir.

Tablo 3.3 Veri setindeki özelliklerin korelasyon analizi.

	Fiyat	Model	Konum	Yıl	Km	Yakıt	Şanz.	Sahibi	Mil.	Motor	Güç	Koltuk
Fiyat	1,00	-0,10	-0,12	0,30	-0,01	-0,30	-0,59	-0,09	-0,34	0,66	0,77	0,06
Model	**-0,10**	1,00	0,02	0,01	0,03	-0,06	0,14	-0,03	-0,04	0,02	-0,15	0,16
Konum	**-0,12**	0,02	1,00	-0,02	-0,05	0,12	0,04	-0,04	0,02	-0,08	-0,07	-0,04
Yıl	0,30	0,01	-0,02	1,00	-0,17	-0,11	-0,09	-0,38	0,29	-0,07	0,01	0,01
Km	**-0,01**	0,03	-0,05	-0,17	1,00	-0,10	0,02	0,08	-0,06	0,09	0,03	0,08
Yakıt	-0,30	-0,06	0,12	-0,11	-0,10	1,00	0,12	0,04	-0,14	-0,40	-0,26	-0,30
Şanz.	-0,59	0,14	0,04	-0,09	0,02	0,12	1,00	0,00	0,37	-0,50	-0,64	0,07
Sahibi	**-0,09**	-0,03	-0,04	-0,38	0,08	0,04	0,00	1,00	-0,14	0,06	0,03	0,00
Mil.	-0,34	-0,04	0,02	0,29	-0,06	-0,14	0,37	-0,14	1,00	-0,64	-0,54	-0,33
Motor	**0,66**	0,02	-0,08	-0,07	0,09	-0,40	-0,50	0,06	-0,64	1,00	0,87	0,40
Güç	**0,77**	-0,15	-0,07	0,01	0,03	-0,26	-0,64	0,03	-0,54	**0,87**	1,00	0,10
Koltuk	**0,06**	0,16	-0,04	0,01	0,08	-0,30	0,07	0,00	-0,33	0,40	0,10	1,00



Şekil 3.4 Veri setindeki verilerin ısı haritası.



Şekil 3.5 Güç ve motor arasındaki ilişki.

Veri seti içerisindeki uç noktaların fazla olması gerçek değerden daha fazla sapmaya sebep olmaktadır. Ayrıca, düşük korelasyon katsayısına sahip girdi ve çıktı parametrelerinin ilişkisi zayıf olacağından, sonucun gerçek değerinden sapma oranı artacaktır. Bu nedenle, modellerin başarısında iyileşme sağlamak için veri seti içerisindeki uç değerler çıkarılmış, sadece yüksek korelasyona sahip özelliklerle model eğitilerek deney tekrarlanmış ve sonuca etkisi araştırılmıştır. Ayrıca aralarında güçlü bir ilişki bulunan güç ve motor ikilisinden (güç ve motor arasındaki r=0.87), gücün aracın fiyat ile olan ilişkisi (güç ve fiyat arasındaki r= 0.77, motor ve fiyat arasındaki r=0.66) daha kuvvetli olduğundan motor özelliğinin veri setinden çıkarılarak modellerin başarısına etkisinin karşılaştırılması için deney yeniden tekrarlanmıştır.

3.3 Veri Ön İşlemesi ve Makine Öğrenmesi

Öncelikle, 11 özellikli (model, konum, yıl, gidilen kilometre, yakıt türü, şanzıman, araç sahibi türü, kilometre, motor, güç, koltuk, fiyat) 5872 otomobil satış veri setine bölüm 3.1'de bahsedilen üç aşamalı bir ön işleme uygulanmıştır. Tüm analizlerde, araç fiyat tahminindeki hatanın hesaplanmasında K-Katlamalı Çapraz Doğrulama yöntemi kullanılmıştır. İlk olarak, k değeri 5 olarak ayarlanmış, veri seti verilen 5 değerine bölündükten sonra, parçalardan biri veriyi test etmek, geriye kalan 4 parça ise veriyi eğitmek için kullanılmıştır. Sonrasında, k değeri 10 olarak ayarlanmış, veri seti verilen 10 değerine bölünmüş ve geriye kalan 9 parça veriyi eğitmek için kullanılmıştır. Daha fazla veri ile eğitilen modellerin hata oranları azalmış ve sonuçlar iyileşmiştir.

Tablo 3.4 Makine Öğrenmesi Yöntemlerinin Sonuçları.

KFOLDCROSS	kf=KFold(10)		
Kodlama Metotları	**Karar Ağacı**	**Rastgele Orman**	**Destek Vektör**
Sıralı (Label)	ort_kfold_rmse: 5.3422 ort_kfold_mape: 21.7830	ort_kfold_rmse: 3.6666 ort_kfold_mape: 16.5605	ort_kfold_rmse: 4.2320 ort_kfold_mape: 22.2190
One Hot	ort_kfold_rmse: 4.5569 ort_kfold_mape: 20.8665	**ort_kfold_rmse: 3.6226** **ort_kfold_mape: 16.1611**	ort_kfold_rmse: 5.9300 ort_kfold_mape: 32.6421
İkili (Binary)	ort_kfold_rmse: 5.0793 ort_kfold_mape: 23.0521	ort_kfold_rmse: 3.7235 ort_kfold_mape: 17.3939	ort_kfold_rmse: 4.2529 ort_kfold_mape: 24.4515
Frekans	ort_kfold_rmse: 5.5721 ort_kfold_mape: 22.8728	ort_kfold_rmse: 3.8224 ort_kfold_mape: 17.1192	ort_kfold_rmse: 4.4110 ort_kfold_mape: 22.6203

KFOLDCROSS	kf=KFold(5)		
Kodlama Metotları	**Karar Ağacı**	**Rastgele Orman**	**Destek Vektör**
Sıralı (Label)	ort_kfold_rmse: 5.3596 ort_kfold_mape: 22.0638	ort_kfold_rmse: 3.9562 ort_kfold_mape: 16.9557	ort_kfold_rmse: 4.3554 ort_kfold_mape: 22.5935
One Hot	ort_kfold_rmse: 5.3288 ort_kfold_mape: 21.2619	**ort_kfold_rmse: 3.8839** **ort_kfold_mape: 16.4136**	ort_kfold_rmse: 6.0304 ort_kfold_mape: 33.6917
İkili (Binary)	ort_kfold_rmse: 5.3389 ort_kfold_mape: 23.4215	ort_kfold_rmse: 3.9272 ort_kfold_mape: 17.7639	ort_kfold_rmse: 4.4168 ort_kfold_mape: 25.1530
Frekans	ort_kfold_rmse: 5.8441 ort_kfold_mape: 22.8227	ort_kfold_rmse: 4.0971 ort_kfold_mape: 17.6195	ort_kfold_rmse: 4.5460 ort_kfold_mape: 22.8294

Diğer aşamada ise sonuçların iyileştirilmesi için verilerin normal dağılıp dağılmadığı incelenmiştir. Şekil 3.1'deki fiyat grafiğinden görüleceği üzere, fiyatı 40 üzeri olan veriler grafiğin normal dağılımını bozmaktadır. Bu yüzden 40'dan yüksek fiyata sahip olan araçların verileri 5872 veri içeren veri setinden çıkarılarak, 5703 veri ile deney tekrarlanmıştır.

Tablo 3.5 Uç Verilerden Arındırılmış Veri Seti İle
Makine Öğrenmesi Yöntemlerinin Sonuçları.

KFOLDCROSS	kf=KFold(10)		
Kodlama Metotları	**Karar Ağacı**	**Rastgele Orman**	**Destek Vektör Makinesi**
Sıralı (Label)	ort_kfold_rmse: 3.0600 ort_kfold_mape: 21.0855	ort_kfold_rmse: 2.1801 ort_kfold_mape: 15.9087	ort_kfold_rmse: 2.4842 ort_kfold_mape: 20.2017
One Hot	ort_kfold_rmse: 3.0529 ort_kfold_mape: 20.0000	**ort_kfold_rmse: 2.2147** **ort_kfold_mape: 15.6661**	ort_kfold_rmse: 3.3879 ort_kfold_mape: 27.5780
İkili (Binary)	ort_kfold_rmse: 3.1628 ort_kfold_mape: 22.0675	ort_kfold_rmse: 2.2843 ort_kfold_mape: 16.7299	ort_kfold_rmse: 2.4716 ort_kfold_mape: 21.6036
Frekans	ort_kfold_rmse: 3.1212 ort_kfold_mape: 21.6549	ort_kfold_rmse: 2.3028 ort_kfold_mape: 16.7324	ort_kfold_rmse: 2.5869 ort_kfold_mape: 20.9895

KFOLDCROSS	kf=KFold(5)		
Kodlama Metotları	**Karar Ağacı**	**Rastgele Orman**	**Destek Vektör Makinesi**
Sıralı (Label)	ort_kfold_rmse: 3.1034 ort_kfold_mape: 21.4711	ort_kfold_rmse: 2.2374 ort_kfold_mape: 16.2197	ort_kfold_rmse: 2.5207 ort_kfold_mape: 20.4385
One Hot	ort_kfold_rmse: 2.8949 ort_kfold_mape: 20.3551	**ort_kfold_rmse: 2.2930** **ort_kfold_mape: 15.9672**	ort_kfold_rmse: 3.4440 ort_kfold_mape: 28.3597
İkili (Binary)	ort_kfold_rmse: 3.2680 ort_kfold_mape: 22.5504	ort_kfold_rmse: 2.3833 ort_kfold_mape: 17.3269	ort_kfold_rmse: 2.5114 ort_kfold_mape: 22.1011
Frekans	ort_kfold_rmse: 3.3021 ort_kfold_mape: 22.4745	ort_kfold_rmse: 2.3609 ort_kfold_mape: 17.1828	ort_kfold_rmse: 2.6331 ort_kfold_mape: 21.1757

3.4 Veri Ön İşlemesinin Çıktılara Etkisinin İncelenmesi

Yapılan deneylerin sonucuna göre veri seti iyileştirildiğinde Tablo 3.12'de görüldüğü üzere makine öğrenmesi yöntemlerinin tahminlemelerindeki hatalarda azalma, sonuçlarda iyileşme gözlemlenmiştir. En iyi sonucu her iki deneyde de rastgele orman makine öğrenmesi yöntemi 10k katlamalı çapraz doğrulama ile vermiştir. Kodlama metotlarının farklılığı sonuçlarda değişikliğe sebep olsa da, bu değişiklik oldukça azdır. Veri iyileştirilmeden önce Ortalama Kare Kök hata metriğine göre one hot kodlama yöntemi en iyi sonucu verirken, veri setinden uç noktalar çıkartıldıktan sonra, sıralı kodlama daha iyi sonuç vermiştir. Ancak, Ortalama Mutlak Yüzde hata metriğine göre one hot kodlama yöntemi her iki durumda da en iyi sonucu vermektedir. Samruddhi ve Kumar [42]'ın uç değerlerden arındırılmamış veri seti üzerinde KNN yöntemini denedikleri çalışmalarında, en başarılı sonucu k-katlamalı değer 10 iken ve K değeri 4 olarak seçildiğinde 4.73 RMSE olarak gözlemlemişlerdir. Veri seti üzerinde yapılan iyileştirmeler sonrasında, bu çalışmada kullanılan diğer makine öğrenmesi yöntemlerinin performansı daha yüksek çıkmıştır.

3.5 YSA Sinir Ağı Optimizasyonu

YSA Sinir Ağı'nda kullanılan parametreler Tablo 3.13'te belirtilmiştir.

Tablo 3.6 Yapay Sinir Ağı (YSA) Parametreleri.

PARAMETRE ADI	PARAMETRE	AÇIKLAMA
hidden_layer_sizes	i	i. gizli katmandaki nöronların sayısını temsil eder.
Activation	"relu"	aktivasyon : {'identification', 'lojistik', 'tanh', 'relu'}, default='relu' Gizli katman için aktivasyon fonksiyonu. - 'identification', işlem gerektirmeyen aktivasyon, doğrusal darboğaz uygulamak için kullanışlıdır, f(x) = x döndürür - 'lojistik', lojistik sigmoid işlevi, f(x) = 1 / (1 + exp(-x)) döndürür. - 'tanh', hiperbolik tan işlevi, f(x) = tanh(x) değerini döndürür. - 'relu', düzeltilmiş doğrusal birim işlevi, f(x) = maks(0, x) döndürür
Solver	'adam'	çözücü : {'lbfgs', 'sgd', 'adam'}, default='adam' Ağırlık optimizasyonu için çözücü. - 'lbfgs', Newton benzeri yöntemler ailesindeki bir optimize edicidir. - 'sgd' stokastik gradyan inişini ifade eder. - 'adam', Kingma, Diederik ve Jimmy Ba tarafından önerilen stokastik gradyan tabanlı bir optimize ediciyi ifade eder. Not: Varsayılan çözücü 'adam', hem eğitim süresi hem de doğrulama puanı açısından nispeten büyük veri kümelerinde (binlerce veya daha fazla eğitim örneğiyle) oldukça iyi çalışır. Ancak küçük veri kümeleri için 'lbfgs' daha hızlı

		birleşebilir ve daha iyi performans gösterebilir.
Alpha	0.0001	alfa : kayan nokta, varsayılan=0,0001 L2, ceza (düzenleme terimi) parametresi.
batch_size	Auto	batch_size : int, default='auto' Stokastik optimize ediciler için mini partilerin boyutu. Çözücü 'lbfgs' ise, sınıflandırıcı minibatch kullanmaz. "auto" olarak ayarlandığında, `batch_size=min(200, n örnek)`
learning_rate	"constant"	öğrenme_hızı : {'sabit', 'invscaling', 'uyarlanabilir'}, varsayılan='sabit' Ağırlık güncellemeleri için öğrenme oranı programı. - 'sabit', başlangıç öğrenme oranı tarafından verilen sabit bir öğrenme oranıdır. - 'ölçeklendirme', 'güç'ün ters bir ölçekleme üssünü kullanarak 't' her zaman adımında 'öğrenme oranını' öğrenme oranını kademeli olarak azaltır. etkin öğrenme oranı = başlangıç öğrenme oranı / pow(t, güç t) - 'uyarlanabilir', eğitim kaybı azalmaya devam ettiği sürece öğrenme oranını başlangıç öğrenme oranı olarak sabit tutar. Ardışık iki dönem eğitim kaybını en az tol azaltamadığında veya 'erken durdurma' açıksa doğrulama puanını en az tol artıramadığında, mevcut öğrenme oranı 5'e bölünür. Yalnızca çözücü='sgd' olduğunda kullanılır.

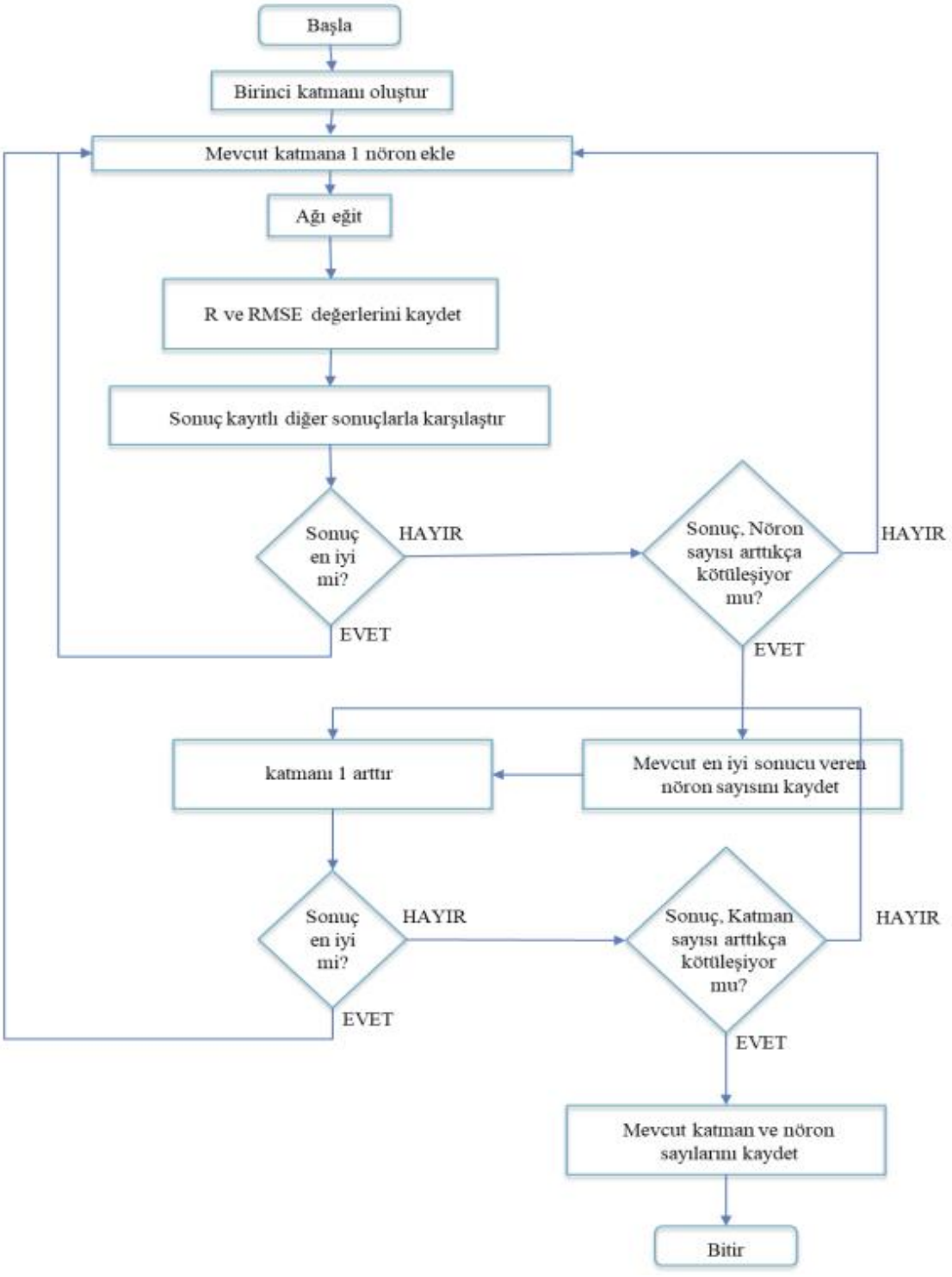
learning_rate_init	0.001	başlangıç öğrenme oranı : çift, varsayılan = 0,001 Kullanılan ilk öğrenme oranı. Ağırlıkların güncellenmesinde adım boyutunu kontrol eder. Yalnızca çözücü='sgd' veya 'adam' olduğunda kullanılır.
power_t	0.5	power_t : çift, varsayılan = 0,5 Ters ölçekleme öğrenme oranı için üs. Learning_rate 'invscaling' olarak ayarlandığında etkin öğrenme oranını güncellemek için kullanılır. Yalnızca çözücü='sgd' olduğunda kullanılır.
max_iter	1000	max_iter : int, varsayılan=200 Maksimum yineleme sayısı. Çözücü, yakınsama ('tol' ile belirlenir) veya bu sayıda yineleme olana kadar yinelenir. Stokastik çözücüler için ('sgd', 'adam'), bunun gradyan adımlarının sayısını değil, dönemlerin sayısını (her bir veri noktasının kaç kez kullanılacağını) belirlediğine dikkat edin.
Shuffle	"True"	shuffle : bool, varsayılan=True Her yinelemede örneklerin karıştırılıp karıştırılmayacağı. Yalnızca çözücü='sgd' veya 'adam' olduğunda kullanılır.
random_state	None	random_state : int, RandomState örneği, varsayılan=Yok Ağırlıklar ve önyargı başlatma için rasgele sayı üretimini, erken durdurma kullanılıyorsa tren testi bölünmesini ve çözücü='sgd' veya 'adam' olduğunda toplu örneklemeyi belirler. Birden çok işlev çağrısında tekrarlanabilir sonuçlar için bir int iletin. Bakınız :term:`Sözlük <random_state>`.

Tol	0,0001	tol : kayan nokta, varsayılan=1e-4 Optimizasyon için tolerans. Kayıp veya puan art arda 'n_iter_no_change' yinelemeleri için en az 'tol' ile iyileşmediğinde, 'öğrenme_hızı' 'uyarlanabilir' olarak ayarlanmadıkça yakınsamaya ulaşıldığı kabul edilir ve eğitim durur.
Verbose	"False"	verbose : bool, varsayılan=Yanlış İlerleme mesajlarının stdout'a yazdırılıp yazdırılmayacağı.
warm_start	"False"	Warm_start : bool, varsayılan=Yanlış True olarak ayarlandığında, başlatma olarak sığdırmak için önceki çağrının çözümünü yeniden kullanın, aksi takdirde önceki çözümü silmeniz yeterlidir. Bakınız :term:`Sözlük <warm_start>`.
momentum	0.9	momentum : kayan nokta, varsayılan=0.9 Gradyan iniş güncellemesi için momentum. 0 ile 1 arasında olmalıdır. Yalnızca çözücü='sgd' olduğunda kullanılır.
nesterovs_momentum	"True"	nesterovs_momentum : boolean, default=True Nesterov'un momentumunun kullanılıp kullanılmayacağı. Yalnızca çözücü='sgd' ve momentum > 0 olduğunda kullanılır.

early_stopping	"True"	Early_stopping : bool, varsayılan=Yanlış Doğrulama puanı iyileşmediğinde eğitimi sonlandırmak için erken durdurmanın kullanılıp kullanılmayacağı. Doğru olarak ayarlanırsa, otomatik olarak eğitim verilerinin %10'unu doğrulama olarak ayıracak ve doğrulama puanı art arda "n_iter_no_change" için en az "tol" artmadığı zaman eğitimi sonlandıracaktır. Yalnızca çözücü='sgd' veya 'adam' olduğunda etkilidir
validation_fraction	0.1	validation_fraction: kayan nokta, varsayılan=0.1 Erken durdurma için doğrulama seti olarak ayrılacak eğitim verilerinin oranı. 0 ile 1 arasında olmalıdır. Yalnızca erken_durdurma doğru ise kullanılır.
beta_1	0.9	beta_1 : kayan nokta, varsayılan=0.9 Adam'daki ilk moment vektörünün tahminleri için üstel bozulma oranı, [0, 1) olmalıdır. Yalnızca çözücü='adam' olduğunda kullanılır
beta_2	0.999	beta_2 : kayan nokta, varsayılan = 0.999 Adam'daki ikinci moment vektörü tahminleri için üstel bozulma oranı, [0, 1) olmalıdır. Yalnızca çözücü='adam' olduğunda kullanılır
Epsilon	1,00E-08	epsilon : kayan nokta, varsayılan=1e-8 Adam cinsinden sayısal kararlılık değeri. Yalnızca çözücü='adam' olduğunda kullanılır
n_iter_no_change	10	n_iter_no_change : int, varsayılan=10 "tol" iyileştirmesini karşılamayan maksimum dönem sayısı. Yalnızca çözücü='sgd' veya 'adam' olduğunda etkilidir

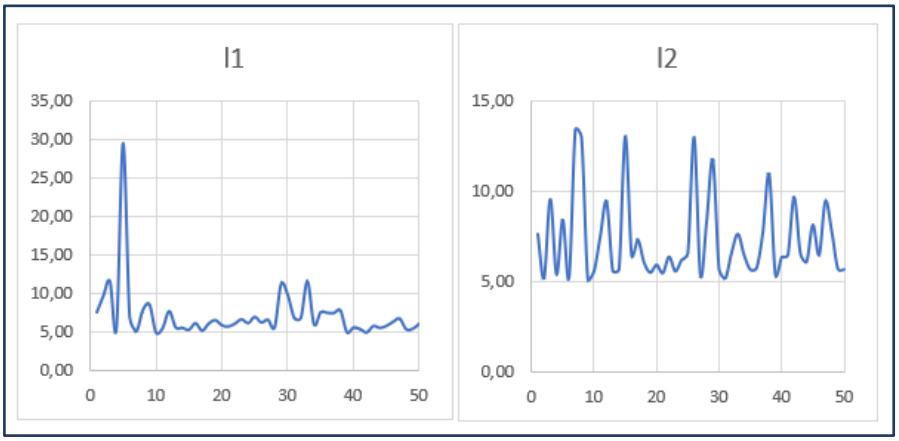
max_fun	15000	max_fun : int, varsayılan=15000 Yalnızca çözücü='lbfgs' olduğunda kullanılır. Maksimum fonksiyon çağrısı sayısı.Çözücü, yakınsama ('tol' ile belirlenir), yineleme sayısı max_iter'e veya bu fonksiyon çağrısı sayısına ulaşana kadar yinelenir. İşlev çağrısı sayısının MLPRegressor için yineleme sayısından büyük veya ona eşit olacağını unutmayın.

Yapay Sinir Ağı Modelindeki optimum katman ve nöron sayılarını bulmak için Şekil 3.6'da gösterildiği gibi bir süreç işletilir. Öncelikle ilk katmana bir nöron eklenir. Başlangıçta bir nöron içeren tek katmanlı bu model, 10-katlamalı çapraz doğrulama yöntemi ile uç değerlerden arındırılmış veri seti ile eğitilir. Eğitim için ayrılan veri seti, tüm veri setinin %90'ı olarak seçilir. Bu eğitim için ayrılan verilerin %10'u da 'random' özelliği kullanılmadan validasyon için kullanılır. Bu süreç, katmandaki nöron sayısı 50'ye ulaşana kadar tekrarlanır. Mevcut katmanda en düşük RMSE ve MAPE değeri veren nöron sayısı bu katman için optimum nöron olarak seçilmektedir. Sisteme bir katman daha eklenerek, ikinci katman için de 50 nörona kadar deneme yapılarak, optimum nöron sayısı bulunmaktadır. Eklenen diğer ikinci katman daha yüksek RMSE ve MAPE değerine neden olursa, modelin başarısı düştüğü için bir önceki katman sistem için optimal katman sayısı olarak belirlenir. Buna karşın, daha düşük RMSE ve MAPE değerine neden olursa, belirlenen optimum nöron sayıları katmanlar için sabit tutularak, katman sayısı arttırılmaya devam edilir.



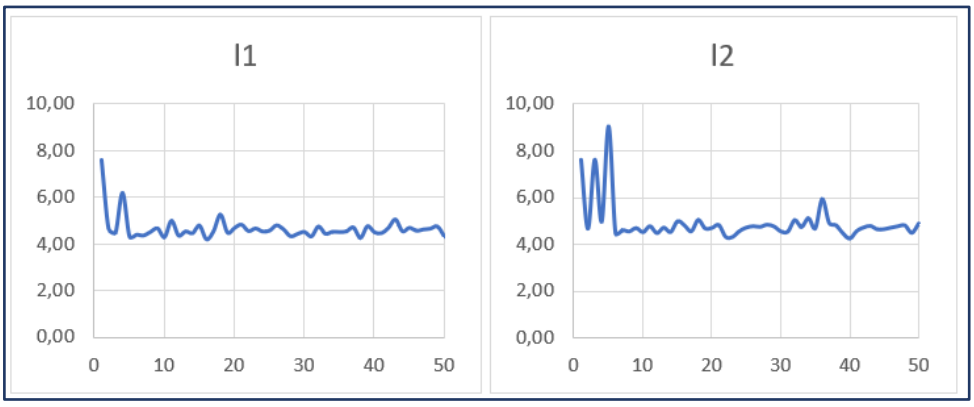
Şekil 3.6 YSA Parametrelerinin Optimizasyonu [41].

3.6 YSA Sinir Ağı Yöntemi Deneyleri ve Sonuçları



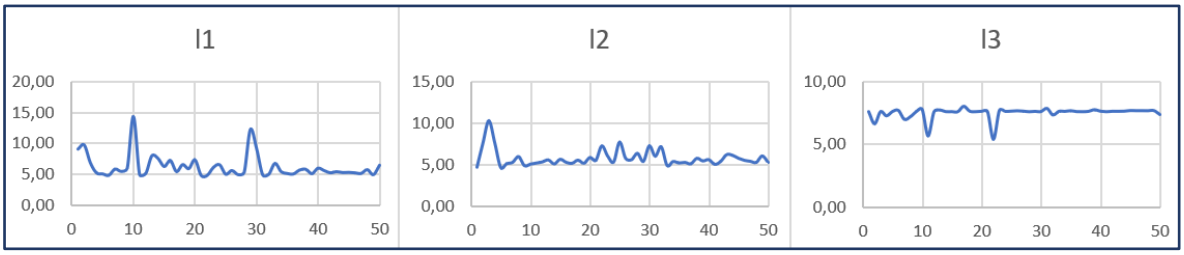
Şekil 3.7 Sıralı Kodlama Metotu Uygulanan Veri Setinin YSA Yöntemi Sonucu Nöron Sayısına Göre L1 ve L2 Katmanlarındaki RMSE Değerleri.

Şekil 3.7'de bir katmanlı olarak eğitilmiş veri setinde 50 nöron içerisinde en düşük RMSE değerinin 5,01 ve nöron sayısının 10 olduğu gözlemlenmiştir. İkinci aşamada ilk katmandaki 10 nöronla sabit tutularak, ikinci katman eklenmiştir ve testin sonucunda ikinci katmanın 5,1 en düşük RMSE değeri ile nöron sayısının 9 iken sağlandığı gözlemlenmiştir. Dolayısıyla, ilk katmanda 10 nöron ile gözlemlenen RMSE değeri daha düşük olduğu için 1 gizli katmanlı sistemin Sıralı Kodlama Metotu uygulanan veri seti için daha ideal olduğu sonucuna varılmıştır. Şekil 3.7'de ikinci katmanda değişen nöron sayısına göre gözlemlenen RMSE değerlerini veren grafikte, grafiğin ikinci katmanla bozulduğu ve daha yüksek değerler verdiği görülmektedir.



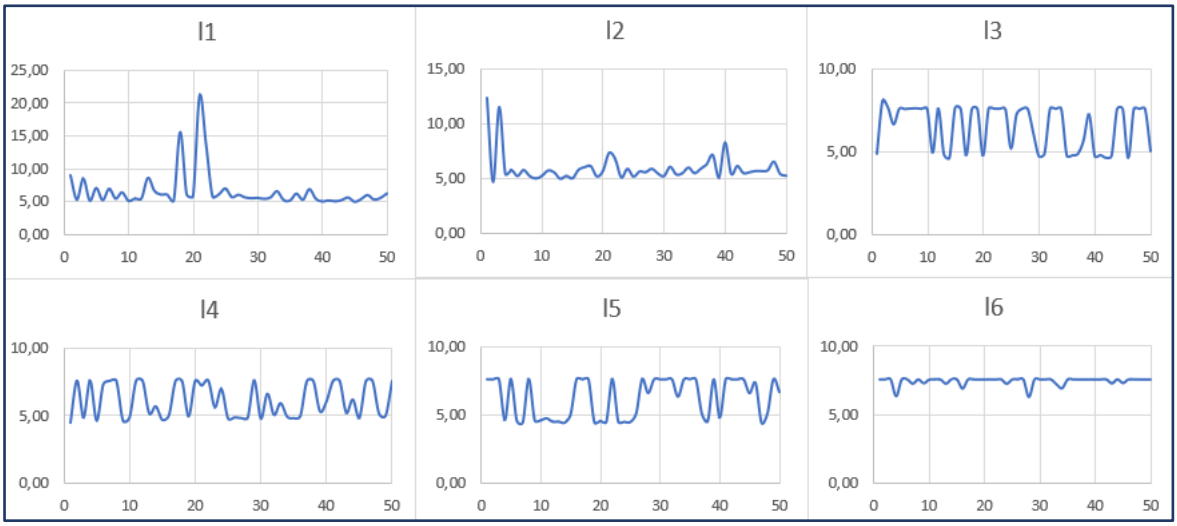
Şekil 3.8 One Hot Kodlama Metotu Uygulanan Veri Setinin YSA Yöntemi Sonucu Nöron Sayısına Göre L1 ve L2 Katmanlarındaki RMSE Değerleri.

Şekil 3.8'de bir katmanlı olarak eğitilmiş veri setinde 50 nöron içerisinde en düşük RMSE değerinin 4,2 ve nöron sayısının 16 olduğu gözlemlenmiştir. Bu nedenle, ikinci katman eklenirken ilk katmandaki 16 nöron sabit tutulmuştur. İkinci katmanda nöron sayısı 40 iken, en düşük RMSE değeri 4,25 olarak gözlemlenmiştir. Dolayısıyla, ilk katmanda 16 nöron ile daha düşük RMSE değeri elde edildiği için, 1 gizli katmanlı sistemin One Hot Kodlama Metotu uygulanan veri seti için optimum olduğu sonucuna varılmıştır.



Şekil 3.9 İkili Kodlama Metotu Uygulanan Veri Setinin YSA Yöntemi Sonucu Nöron Sayısına Göre L1, L2 ve L3 Katmanlarındaki RMSE Değerleri.

İkili Kodlama Metotu uygulanan veri seti bir katmanlı olarak eğitilmiş, ve 50 nöron içerisinde en düşük RMSE değerinin 4,86 nöron sayısının 21 olduğu gözlemlenmiştir. İkinci aşamada ilk katman 21 nöronla sabit tutularak, ikinci katman eklenmiştir ve testin sonucunda ikinci katman 4,61 en düşük RMSE değeri ile nöron sayısı 5 iken sağlanmıştır. Bu değer, tek katmanlı modelin RMSE değerinden (4,86) daha düşük olduğundan, üçüncü katmanın eklenmsine karar verilmiştir. Üçüncü katman eklenirken, daha önce olduğu gibi önceki katmanlardaki nöron sayıları sabit tutulmuştur (21 ve 5). Testler sonucunda 22 nöronla en düşük RMSE değeri 5,45 olarak gözlemlendiğinden ve bu değer iki katmanlı sisteme göre daha yüksek olduğundan, Sıralı Kodlama Metotu uygulandığı zaman iki katmanlı YSA modelinin ideal olduğu sonucuna varılmıştır. Şekil 3.9'da ikinci katmanda değişen nöron sayısına göre gözlemlenen RMSE değerleri 5 civarında iken, üçüncü katman eklendiği zaman RMSE ortalamasının artarak 7 civarında yer aldığı görülmektedir.



Şekil 3.10 Frekans Kodlama Metotu Uygulanan Veri Setinin YSA Yöntemi Sonucu Nöron Sayısına Göre L1, L2, L3, L4, L5 ve L6 Katmanlarındaki RMSE Değerleri.

Şekil 3.10'da Frekans Kodlama Metotu uygulanan veri seti, bir katmanlı olarak eğitildiği zaman 50 nöron içerisinde en düşük RMSE değerinin 4,96 ve nöron sayısının 45 olduğu gözlemlenmiştir. 6. katmana kadar eklenen katmanlarda nöron sayıları değiştikçe, daha düşük RMSE değerleri gözlemlenmiştir. Ancak 6. katmana gelindiği zaman hem RMSE değerinin 50 nöron için ortalaması artmış, hem de nöronlar değiştikçe daha düşük RMSE değerine rastlanamamıştır. Dolayısıyla, beş katmanlı Yapay Sinir Ağı yönteminin Frekans Kodlama Metotu uygulanan veri seti için en ideal olduğu sonucuna varılmıştır.

Sonuç olarak, Tablo 3.7'de belirtildiği üzere Yapay Sinir Ağı yöntemi One Hot Kodlama Metotu uygulanan veri setinde en iyi sonucu vermektedir.

Tablo 3.7 Yapay Sinir Ağı (YSA) Sonucu

Nöron Sayısına Göre L1, L2, L3, L4, L5 ve L6 Katmanlarındaki RMSE Değerleri.

KFOLDCROSS **kf=KFold(10)** early_stopping is true max iter: 1000 epsilon=1e-8

Kodlama Metotları	Birinci Katman			İkinci Katman		
	nöron sayısı	ort	rmse	nöron sayısı	ort	rmse
Sıralı (Label)	10	7,2	5,01	9	7,28	5,1
One Hot	16	4,67	4,2	40	4,93	4,25
İkili (Binary)	21	6,22	4,86	5	5,74	4,61
Frekans	45	6,53	4,96	2	5,97	4,75

Kodlama Metotları	Üçüncü Katman			Dördüncü Katman		
	nöron sayısı	ort	rmse	nöron sayısı	ort	rmse
Sıralı (Label)	24	6,21	5,08	15	6,37	5,18
One Hot	41	5,1	4,13	10	5,01	4,27
İkili (Binary)	22	7,48	5,45	6	7,77	6,7
Frekans	42	6,48	4,62	1	6,14	4,51

Kodlama Metotları	Beşinci Katman			Altıncı Katman		
	nöron sayısı	ort	rmse	nöron sayısı	ort	rmse
Sıralı (Label)						
One Hot						
İkili (Binary)						
Frekans	47	6,23	4,45	28	7,49	6,33

YSA modeli deneyinin ikinci aşamasında 5703 otomobil satış veri setinden fiyat ile korelasyon oranı düşük olan araç sahibi türü, kilometre ve koltuk özellikleri çıkarılmıştır.

Ek olarak, aralarında güçlü bir ilişki bulunan güç ve motor ikilisinden gücün aracın fiyatı ile olan ilişkisi daha kuvvetli olduğundan motor özelliği veri setinden çıkarılarak 8 özellikli veri seti üzerinde daha önce 4,2 ile en iyi sonucu aldığımız one hot kodlaması kullanılarak deney yeniden tekrarlanmıştır. Tablo 3.8'deki gibi sonuçlarda iyileşme gözlemlenmiştir.

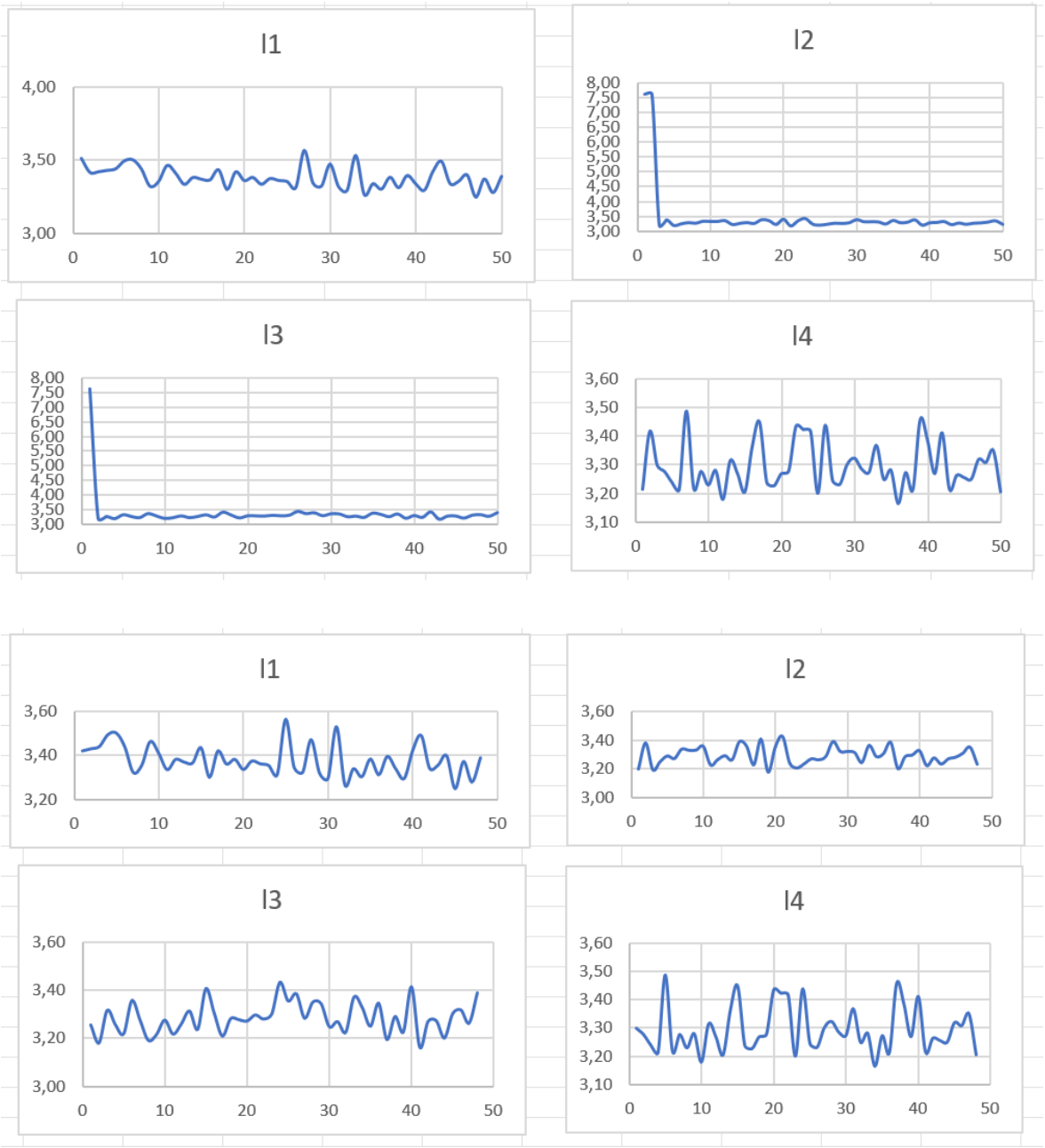
Tablo 3.8 8 Özellikli Veri Setinin Yapay Sinir Ağı (YSA) Sonucu

Nöron Sayısına Göre L1, L2, L3 ve L4 Katmanlarındaki RMSE Değerleri.

KFOLDCROSS **kf=KFold(10)** early_stopping is true max iter: 1000 epsilon=1e-8

Kodlama Metotları	Birinci Katman			İkinci Katman		
	nöron sayısı	ort	rmse	nöron sayısı	ort	rmse
One Hot	47	3,38	3,25	21	3,58	3,18

Kodlama Metotları	Üçüncü Katman			Dördüncü Katman		
	nöron sayısı	ort	rmse	nöron sayısı	ort	rmse
One Hot	43	3,37	3,16	37	3,3	3,17



Şekil 3.11 8 Özellikli Veri Setinin YSA Yöntemi Sonucu
Nöron Sayısına Göre L1, L2, L3 ve L4 Katmanlarındaki RMSE Değerleri (üstteki grafikler), 3.00-3.60 rmse değerleri aralığının büyütülmüş hâli (alttaki grafikler).

Şekil 3.11'de bir katmanlı olarak eğitilmiş veri setinde 50 nöron içerisinde en düşük RMSE değerinin 3,25 ve nöron sayısının 47 olduğu gözlemlenmiştir. Bu nedenle, ikinci katman eklenirken ilk katmandaki 47 nöron sabit tutulmuştur. İkinci katmanda nöron sayısı 21 iken, en düşük RMSE değeri 3,18 olarak gözlemlenmiştir. Dolayısıyla, ikinci katmanda 21 nöron ile daha düşük RMSE değeri elde edildiği için, üçüncü katman eklenmiştir. 43

nöron sayısıyla 3,16 RMSE değeri elde edilmiştir. Dördüncü katmanın eklenmesi ile en düşük RMSE değerinin 3,17 ve nöron sayısının 37 olduğu gözlemlenmiştir. Hata değeri artmaya başladığı için, üç gizli katmanlı sistemin One Hot Kodlama Metotu uygulanan bu veri seti için en iyi olduğu sonucuna varılmıştır.

3.7 Veri Setindeki Değişkenlerin Fiyat Tahminine Etkisinin Değerlendirilmesi

Yapılan istatistiksel çalışmaların sonuçları ile, makine öğrenimi yöntemlerinin sonuçları karşılaştırmak üzere uç değerlerden arındırılmış veri seti içerisindeki 11 özelliğin her birinin veri setinden çıkarılıp kalan 10 özellikle ile yapılan tahminlemenin sonuçlara etkisi araştırılmıştır. Önceki deneylerde en iyi sonuçlar, One Hot Kodlama Metotu ile alındığı için, değişken etkisi analizinde de bu kodlama yöntemi kullanılmıştır. Tablo 3.9'da veri setinden çıkarılan her bir özellik sonucunda elde edilen 10 özellikli veri seti ile fiyat tahminindeki hata değerleri (RMSE ve MAPE) yer almaktadır.

Tablo 3.9'daki sonuçlardan da görüleceği üzere hata değerleri genellikle artmaktadır. Bunun nedeni, her özelliğin Tablo 3.10'da gösterildiği gibi fiyat ile bir ilişkisinin olmasıdır. Hata değerleri ne kadar artarsa, o özelliğin fiyat tahminine etkisi o kadar fazla demektir. Bu bağlamda, her makine öğrenmesi tekniği içi ayrı ayrı olmak üzere özelliklerin sonuca etkisi en azdan (RMSE değeri en düşük) en çoğa (RMSE değeri en yüksek) göre sıralanmıştır. Tablo 3.10'a göre fiyat ile arasında en yüksek korelasyon bulunan "Güç" özelliğinin, bütün makine öğrenimi teknikleri sonuçlarına göre de en belirleyici 2. özellik olduğu gözlenmiştir. Öte yandan, Tablo 3.10'a göre fiyat üzerinde neredeyse hiçbir etkisi olmayan (-0,01 korelasyon değeri) "Kilometre" özelliği de Destek Vektör Makinesi hariç diğer tüm yöntemlerde en az belirleyici özellik olarak tespit edilmiştir. Hatta, bu özelliğin çıkarılması ile daha düşük RMSE ile tahmin yapılabildiğinden, makine öğrenimi yöntemlerinin başarılarının arttığı da görülmektedir. Bu açıdan, Tablo 3.10'da verilen özellikler ile fiyat arasındaki korelasyon değerlerinin Tablo 3.9'daki sonuçlar ile tutarlı olduğu söylenebilir.

Tablo 3.9 Veri setinden çıkarılan özelliklerin deney sonucuna etkisi

KFOLDCROSS	kf=KFold(10)
Özellikler	**Karar Ağacı**
Kilometre	ort_kfold_rmse: 2.8858 ort_kfold_mape: 18.9322
Şanzıman	ort_kfold_rmse: 2.9650 ort_kfold_mape: 19.9629
Motor	ort_kfold_rmse: 2.9733 ort_kfold_mape: 20.0794
Mil	ort_kfold_rmse: 2.9935 ort_kfold_mape: 20.0467
Koltuk	ort_kfold_rmse: 2.9953 ort_kfold_mape: 19.9853
Araç Sahibi Türü	ort_kfold_rmse: 2.9969 ort_kfold_mape: 20.0740
Konum	ort_kfold_rmse: 3.0011 ort_kfold_mape: 22.0537
Yakıt Tipi	ort_kfold_rmse: 3.0270 ort_kfold_mape: 20.0142
Model	ort_kfold_rmse: 3.1254 ort_kfold_mape: 20.8029
Güç	ort_kfold_rmse: 3.3081 ort_kfold_mape: 21.7287
Yıl	ort_kfold_rmse: 3.8208 ort_kfold_mape: 27.9419
11 Özellikli Veri Seti	ort_kfold_rmse: 3.0529 ort_kfold_mape: 20.0000

KFOLDCROSS	kf=KFold(10)
Özellikler	**Rastgele Orman**
Kilometre	ort_kfold_rmse: 2.2117 ort_kfold_mape: 15.2481
Araç Sahibi Türü	ort_kfold_rmse: 2.2145 ort_kfold_mape: 15.6584
Yakıt Tipi	ort_kfold_rmse: 2.2149 ort_kfold_mape: 15.7037
Koltuk	ort_kfold_rmse: 2.2246 ort_kfold_mape: 15.6452
Mil	ort_kfold_rmse: 2.2451 ort_kfold_mape: 15.9083
Model	ort_kfold_rmse: 2.2464 ort_kfold_mape: 16.0166
Şanzıman	ort_kfold_rmse: 2.2493 ort_kfold_mape: 15.8054
Motor	ort_kfold_rmse: 2.2528 ort_kfold_mape: 15.8908
Konum	ort_kfold_rmse: 2.3066 ort_kfold_mape: 17.7648
Güç	ort_kfold_rmse: 2.5700 ort_kfold_mape: 17.4202
Yıl	ort_kfold_rmse: 2.8500 ort_kfold_mape: 22.0264
11 Özellikli Veri Seti	ort_kfold_rmse: 2.2147 ort_kfold_mape: 15.6661

KFOLDCROSS	kf=KFold(10)
Özellikler	**Destek Vektör Makinesi**
Model	ort_kfold_rmse: 2.6641 ort_kfold_mape: 21.6728
Koltuk	ort_kfold_rmse: 3.3801 ort_kfold_mape: 27.4752
Mil	ort_kfold_rmse: 3.3980 ort_kfold_mape: 27.4340
Araç Sahibi Türü	ort_kfold_rmse: 3.3998 ort_kfold_mape: 27.5326
Motor	ort_kfold_rmse: 3.4099 ort_kfold_mape: 27.0837
Kilometre	ort_kfold_rmse: 3.4198 ort_kfold_mape: 27.7260
Yakıt Tipi	ort_kfold_rmse: 3.4301 ort_kfold_mape: 27.2359
Konum	ort_kfold_rmse: 3.4677 ort_kfold_mape: 29.3081
Şanzıman	ort_kfold_rmse: 3.5142 ort_kfold_mape: 28.0168
Güç	ort_kfold_rmse: 3.5189 ort_kfold_mape: 27.9656
Yıl	ort_kfold_rmse: 3.5975 ort_kfold_mape: 30.4254
11 Özellikli Veri Seti	ort_kfold_rmse: 3.3879 ort_kfold_mape: 27.5780

KFOLDCROSS	kf=KFold(10)
Özellikler	**Birinci katman, nöron sayısı=16 Yapay Sinir Ağları**
Kilometre	ort_kfold_rmse: 3,6294
Araç Sahibi Türü	ort_kfold_rmse: 4,3992
Yıl	ort_kfold_rmse: 4,4075
Yakıt Tipi	ort_kfold_rmse: 4,4566
Koltuk	ort_kfold_rmse: 4,4925
Mil olarak alınan yol	ort_kfold_rmse: 4,5358
Motor	ort_kfold_rmse: 4,5561
Şanzıman	ort_kfold_rmse: 4,8426
Konum	ort_kfold_rmse: 4,8772
Güç	ort_kfold_rmse: 5,0252
Model	ort_kfold_rmse: 5,4285
11 Özellikli Veri Seti	ort_kfold_rmse: 4,1962

Tablo 3.10 Özellikler ile Fiyat Arasındaki Korelasyon

Korelasyon Tablosu

Özellikler	Fiyat
Kilometre	-0,01
Koltuk	0,06
Araç Sahibi Türü	-0,09
Model	-0,10
Konum	-0,12
Yıl	0,30
Yakıt Tipi	-0,30
Mil olarak alınan yol	-0,34
Şanzıman	-0,59
Motor	0,66
Güç	0,77

4. SONUÇ VE ÖNERİLER

Otomotiv sektöründe araç fiyatı tahmini yapıldığı zaman göz önünde bulundurulması gereken çok fazla özellik bulunmaktadır. Bu çalışmada, makine öğrenmesi yöntemleri kullanılarak, tahmin sürecinin daha hızlı ve kolay olması sağlanmaya çalışılmıştır.

Veri seti içerindeki özelliklerin birbiriyle olan korelasyonu ve normal dağılımı dikkate alınarak uç değerler çıkartıldığı zaman makine öğrenmesi modellerinin tahmin etme performansının da artacağı gözlemlenmiştir. Kodlama yöntemleri arasında, One Hot kodlama yöntemi bütün modellerde en iyi sonucu vermiştir.

Makine öğrenmesi yöntemlerinin tahmin etme performansları birbirlerine çok yakın olsa da, Rastgele Orman Regresyon yöntemi bu çalışmada kullanılan diğer yöntemler arasında en iyi sonucu vermiştir. Yapay Sinir Ağı yönteminin tahmin etme performansı ise, diğer makine öğrenmesi yöntemlerine göre en düşük çıkmıştır. Normal dağılım göstermeyen bu veri seti, uç değerlerden arındırıldığında ise RMSE değerinin daha düşük çıktığı ve dolayısı ile de Yapay Sinir Ağı yönteminin tahminleme performansının arttığı gözlemlenmiştir. Özelliklerin etkisinin araştırıldığı çalışmada ise veri setinden çıkarılan özelliklerin modellerin performansını ne kadar düşürdüğü gözlemlenmiştir. Güç özelliğinin her tabloda fiyata etki eden ilk üç özellik arasında yer aldığı görülmüş ve fiyatı belirlemede en etkili özellikler arasında olduğu çıkarımı yapılmıştır. Ayrıca, kilometre özelliği veri setinden çıkarıldığı zaman, model sonuçlarında etkisinin olmadığı; hatta bazı modellerde kısmen iyileşme sağladığı sonucuna varılmıştır.

Bu çalışmada kullanılan makine öğrenmesi yöntemlerine ek olarak derin öğrenme yöntemleri ile, daha büyük veri seti kullanılarak sistemin tahminleme yeteneğinin daha iyi olacağı beklenmektedir.

KAYNAKLAR

[1] Mordor Intelligence. (2021) Used car market - growth, trends, COVID-19 impact, and forecasts (2021-2026) [Online]. Mordorintelligence.com.
Available:
mordorintelligence.com/industry-reports/global-used-car-market-growth-trends-and-forecast-2019-2024.

[2] X. Chen, S. Gu, X. Deng, L. Huang, "Used Car Prices in India: What about Future?" *Proceedings of the 7th ICFIED 2022 / Advances in Economics, Business and Management Research*, vol. 648, pp.831-840, 2022.

[3] N. Monburinon, P. Chertchom, T. Kaewkiriya, S. Rungpheung, S. Buya, P. Boonpou, "Prediction of Prices for Used Car by Using Regression Models," *IEEE, 5th International Conference on Business and Industrial Research (ICBIR),* pp.115-119, 2018.

[4] H. Daştan, "Determination of The Factors That Effect Second-Hand Automobile Prices in Turkey by Using Hedonic Pricing Model (in Turkish)," *Gazi Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi,* 1(18), pp.303-327, 2016.

[5] W. Emons, ve G. Sheldon, "The market for used cars: new evidence of the lemons phenomenon," *Applied Economics.*, vol. 41, no. 22, pp. 2867-2885, Oct. 2009.

[6] E. Gegic, B. Isakovic, D. Keco, Z. Masetic, J. Kevric, "Car Price Prediction using Machine Learning Techniques," *TEM Journal.*, vol. 8, pp. 113-118, February 2019, doi: 10.18421/TEM81-16.

[7] L. Mariana, "Support vector regression analysis for price prediction in a car leasing application," Doctoral dissertation, Master thesis, TU Hamburg-Harburg, 2009.

[8] M. M. Karakoç, G. Çelik ve A. Varol, "Car Price Prediction Using An Artificial Neural Network," *Eastern Anatolian Journal of Science*, vol. 6, pp. 44-48, 2020

[9] P. Gajera, A. Gondaliya, J. Kavathiya, “Old Car Price Prediction With Machine Learning,” *International Research Journal of Modernization in Engineering Technology and Science*, vol. 3, pp. 284-290, March 2021

[10] K. Noor, S. Jan, ”Vehicle Price Prediction System using Machine Learning Techniques,” *International Journal of Computer Applications,* vol. 67, pp. 27-31, June 2017

[11] S. Pudaruth, “Predicting the Price of Used Cars Using Machine Learning Techniques,” *International Journal of information & Computation Technology,* pp.753-764, 2014

[12] N. Pal, P. Arora, D. Sundararaman, P. Kohli, S. S. Palakurthy, “How much is my car worth? A methodology for predicting used cars prices using Random Forest,” *Future of Information and Communications Conference*, pp. 1-6, 2018

[13] Madhuvanthi.K, Nallakaruppan.M.K, Senthilkumar N C, Siva Rama Krishnan S, “Car Sales Prediction Using Machine Learning Algorithmns,” *International Journal of Innovative Technology and Exploring Engineering*, pp. 1039-1050, March 2019

[14] Z. Chen, C. Li, W. Sun, “Bitcoin price prediction using machine learning: An approach to sample dimension engineering,” *Journal of Computational and Applied Mathematics*, pp. 1-13, 2020

[15] A. B. Adetunji, O. Noah Akande, F. Alaba Ajala, O. Oyewo, Y. F. Akande, G. Oluwadara, “House Price Prediction using Random Forest Machine Learning Technique,” *The 8th International Conference on Information Technology and Quantitative Management (ITQM 2020 & 2021) / Procedia Computer Science*, vol. 200, pp. 806-813, 2022

[16] X. Zhou, P. Lu, Z. Zheng, D. Tolliver, A. Keramati, “Accident Prediction Accuracy Assessment for Highway-Rail Grade Crossings Using Random Forest Algorithm Compared with Decision Tree,” *ScienceDirect / Reliability Engineering and System Safety,* vol. 200, pp. 1-9, 2020

[17] A. Zeng, S. Liu, Y. Yu, "Comparative study of data driven methods in building electricity use prediction," *ScienceDirect / Energy & Buildings,* vol. 194, pp. 289-300, 2019

[18] P. Sonar, Prof. K. JayaMalini, "Diabetes Prediction Using Different Machine Learning Approaches," *Proceedings of the Third International Conference on Computing Methodologies and Communication (ICCMC),* pp. 367-371, 2019

[19] A.L.D. Loureiro, V.L. Miguéis, Lucas F.M. da Silva, "Exploring the use of deep neural networks for sales forecasting in fashion retail," *ScienceDirect / Decision Support Systems,* vol. 114, pp. 81-93, 2018

[20] S. Vhatkar, J. Dias, "Oral-Care Goods Sales Forecasting Using Artificial Neural Network Model," *7th International Conference on Communication, Computing and Virtualization / Procedia Computer Science,* vol. 79, pp. 238-243, 2016

[21] A. Balogun, A. Tella, "Modelling and investigating the impacts of climatic variables on ozone concentration in Malaysia using correlation analysis with random forest, decision tree regression, linear regression, and support vector regression," *Chemosphere / ScienceDirect,* vol. 299, pp. 1-11, 2022

[22] J. Dou, A. P. Yunus, D. T. Bui, A. Merghadi, M. Sahana, Z. Zhu, C. Chen, K. Khosravi, Y. Yang, B. T. Pham, "Assessment of advanced random forest and decision tree algorithms for modeling rainfall-induced landslide susceptibility in the Izu-Oshima Volcanic Island, Japan," *Science of the Total Environment / ScienceDirect,* vol. 662, pp. 332-346, 2019

[23] P. R. Kadavi, C. Lee, S. Lee, "Landslide-susceptibility mapping in Gangwon-do, South Korea, using logistic regression and decision tree models," *Environmental Earth Sciences,* pp. 78-116, 2019

[24] A. J. Myles, R. N. Feudale, Y. Liu, N. A. Woody, and S. D. Brown, "An introduction to decision tree modeling," *Journal of Chemometrics,* vol. 18, no. 6, pp. 275–285, 2004

[25] D. T. Bui, B. Pradhan, O. Lofman, and I. Revhaug, “Landslide Susceptibility Assessment in Vietnam Using Support Vector Machines, Decision Tree, and Naıve Bayes Models,” *Hindawi Publishing Corporation Mathematical Problems in Engineering*, pp. 1-26, 2012, doi:10.1155/2012/974638

[26] G. K. F. Tso and K. K. W. Yau, “Predicting electricity energy consumption: a comparison of regression analysis, decision tree and neural networks,” Energy, vol. 32, no. 9, pp. 1761–1768, 2007

[27] Curram SP, Mingers J. Neural networks, “Decision tree induction and discriminant analysis: an empirical comparison,” Journal Of The Operational Research Society, vol. 45, pp. 440-450, 1994

[28] Y. Zhao and Y. Zhang, “Comparison of decision tree methods for finding active objects,” Advances in Space Research, vol. 41, no. 12, pp. 1955–1959, 2008

[29] K. Sadia, R. Reza, A. Alam, M. A. Rahman, “Car Parking Availability Prediction: A Comparative Study of LSTM and Random Forest Regression Approaches,” Int J Auto AI Mach Learn. pp. 16-29, 2021

[30] Kuhn, M. Johnson, Kjell, “Applied Predictive Modeling” New York: Springer, 2018

[31] C. Cortes, V. Vapnik, “Support-vector networks”, Mach. Learn., vol. 20, pp. 273–297, 1995

[32] M. Wauters, M. Vanhoucke, “Support Vector Machine Regression for project control forecasting,” Automation in Construction, vol. 47, pp. 92-106, November 2014

[33] Graupe, D. “Principles of artificial neural networks. Advanced series on circuits and systems,” World Scientific Publishing, Singapore City, vol.6, 2007

[34] M. B Patel, S. R Yalamalle, “Stock Price Prediction Using Artificial Neural

Network," International Journal of Innovative Research in Science, Engineering and Technology, vol. 3, pp. 13755- 13762, June 2014

[35] K.K.Sureshkumar, Dr.N.M.Elango, "Performance Analysis of Stock Price Prediction using Artificial Neural Network," Global Journal of Computer Science and Technology, vol. 12, pp. 19-25, January 2012

[36] A. İşeri, B. Karlık, "An artificial neural networks approach on automobile pricing," Expert Systems with Applications / Science Direct, vol. 36, pp. 2155–2160, 2009

[37] Versace, M., Bhatt, R., Hinds, O., & Shiffer, M. "Predicting the exchange traded fund DIA with a combination of genetic algorithms and neural networks," Expert Systems with Applications, vol. 27, pp. 417–425, 2004

[38] Perwej, Y., & Perwej, A. "Prediction of the Bombay Stock Exchange (BSE) market returns using artificial neural network and genetic algorithm," Journal of Intelligent Learning Systems and Applications, vol. 4, pp. 108–119, 2012

[39] Subasi, A., & Erçelebi, E., "Classification of EEG signals using neural network and logistic regression," Computer Methods and Programs in Biomedicine, vol. 78, pp. 87–99, 2005

[40] Kuruş, O. A., Kılıç, N., ve Uçan, O. N., "Hermitian transform approach in classification of ECG signals," Istanbul Aydın Üniversitesi Dergisi, vol. 2, pp. 89-101, 2013

[41] N. U. Aktan, "Havacılık Sanayinde Kullanılan Takım Ve Aparatların Tasarım Sürelerinin Makine Öğrenmesi Yöntemleri İle Kestirilmesi," M.S. Thesis, Dept. Computer. Eng., Başkent Univ., Ankara, Türkiye, 2020

[42] K. Samruddhi¸ Dr. R. Ashok Kumar, "Used Car Price Prediction using K-Nearest Neighbor Based Model," International Journal of Innovative Research in Applied

Sciences and Engineering (IJIRASE), vol. 4, pp. 686-689, September 2020, doi: 10.29027/IJIRASE.v4.i3.2020.686-689

[43] A. Kasliwal.” Used Cars Price Prediction.” Kaggle.com. kaggle.com/datasets/avikasliwal/used-cars-price-prediction?select=train-data.csv.