

**MARMARA ÜNİVERSİTESİ**

**TEKNOLOJİ FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**BİTİRME PROJESİ**

Web Scraping ve Makine Öğrenmesi ile Araç Fiyat Tahmini

**PROJE YAZARI**

Ali Mert Koç

**DANIŞMAN**

Doç. Dr. Kazım Yıldız

**İstanbul, 2022**



**MARMARA ÜNİVERSİTESİ**

**TEKNOLOJİ FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**BİTİRME PROJESİ**

Web Scraping ve Makine Öğrenmesi ile Araç Fiyat Tahmini

**PROJE YAZARI**

Ali Mert Koç

170418018

**DANIŞMAN**

Doç. Dr. Kazım Yıldız

**İstanbul, 2022**

**MARMARA ÜNİVERSİTESİ**

**TEKNOLOJİ FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

Marmara Üniversitesi Teknoloji Fakültesi Bilgisayar Mühendisliği Öğrencisi Ali Mert Koç’un “Web Scraping ve Makine Öğrenmesi ile Araç Fiyat Tahmini” başlıklı bitirme projesi çalışması, 08/06/2022 tarihinde sunulmuş ve jüri üyeleri tarafından başarılı bulunmuştur.

**Jüri Üyeleri**

Prof. Dr. Adı SOYADI (Danışman)

Marmara Üniversitesi ......................................................................... (İMZA) ..................

Doç. Dr. Adı SOYADI (Üye)

Marmara Üniversitesi ......................................................................... (İMZA) ..................

Dr. Öğr. Üyesi Adı SOYADI (Üye)

Marmara Üniversitesi ......................................................................... (İMZA) ..................

**İÇİNDEKİLER**

[SEMBOLLER LİSTESİ / SYMBOLS LIST 6](#_Toc104929007)

[KISALTMALAR/ABBREVIATIONS 7](#_Toc104929008)

[ŞEKİL LİSTESİ 8](#_Toc104929009)

[TABLO LİSTESİ 9](#_Toc104929010)

[ÖZET 10](#_Toc104929011)

BÖLÜM [1. GİRİŞ 11](#_Toc104929012)

[1.1. Giriş 11](#_Toc104929013)

[1.1.1. Problemin tanımı 12](#_Toc104929014)

[1.1.2. Amaç ve hedef 12](#_Toc104929015)

[1.1.3. Ana katkılar 12](#_Toc104929016)

[1.1.4. Tez organizasyonu 12](#_Toc104929017)

[1.2. Web Scraping Nedir? 13](#_Toc104929018)

[1.2.1. Web Parsing (ayrıştırmak) nedir? 13](#_Toc104929019)

[1.2.2. HTTP isteği ve HTML yanıtı 14](#_Toc104929020)

[1.2.3. Python nedir? 15](#_Toc104929021)

[1.2.3.1. Google Colaboratory 15](#_Toc104929022)

[1.2.3.2. BeautifulSoup4 Kütüphanesi 16](#_Toc104929023)

[1.2.3.3. Requests Kütüphanesi 16](#_Toc104929024)

[1.2.3.4. csv Kütüphanesi 17](#_Toc104929025)

[1.2.3.5. Numpy Kütüphanesi 18](#_Toc104929026)

[1.2.3.6. Pandas Kütüphanesi 19](#_Toc104929027)

[1.2.3.7. Seaborn Kütüphanesi 19](#_Toc104929028)

[1.2.3.8. Matplotlib Kütüphanesi 19](#_Toc104929029)

[1.2.3.9. Scikit-learn Kütüphanesi 19](#_Toc104929030)

[1.2.3.10. statsmodels Kütüphanesi 19](#_Toc104929031)

[1.3. Makine öğrenmesi 20](#_Toc104929032)

[1.3.1. Makine öğrenmesi algoritmaları 20](#_Toc104929033)

[1.3.1.1. Linear Regression (Lineer Regresyon) 20](#_Toc104929034)

[1.3.1.2. Support Vector Machines (SVM) 21](#_Toc104929035)

[1.3.1.3. Random Forest 22](#_Toc104929036)

[1.3.1.4. Decision Trees 23](#_Toc104929037)

[1.4. Sınıflandırma 24](#_Toc104929038)

[1.5. Literatür Çalışmaları 25](#_Toc104929039)

BÖLÜM [2.MATERYAL VE YÖNTEM 29](#_Toc104929040)

[2.1. Veriseti elde edinimi 29](#_Toc104929041)

[2.1.1. Websitesinden veri kazıma süreci 29](#_Toc104929042)

[2.1.2. Proxy Rotating işlemi 30](#_Toc104929043)

[2.1.3. Kazımı yapılan verinin verisetine dönüştürülme süreci 30](#_Toc104929044)

[2.2. Verisetinin işlenmesi 30](#_Toc104929045)

[2.2.1. OLS (Ordinary Least Squares, Sıradan En Küçük Kareler Yöntemi) uygulanması 31](#_Toc104929046)

[2.2.2. P Değeri uygulanması 31](#_Toc104929047)

[2.2.3. Model Eğitimi 31](#_Toc104929048)

[2.2.3.1. Cross Validation 31](#_Toc104929049)

BÖLÜM [3. BULGULAR VE TARTIŞMA 31](#_Toc104929050)

[3.1. Deney Ortamı 32](#_Toc104929051)

[3.2. Performans Metrikleri 32](#_Toc104929052)

[3.3. Deney Sonuçları 32](#_Toc104929053)

[3.3.1. Doğrusal Regresyon Sonuçları 33](#_Toc104929054)

[3.3.2. Decision Tree Sonuçları 33](#_Toc104929055)

[3.3.3. Random Forest Sonuçları 33](#_Toc104929056)

[3.3.4. Sonuçların Karşılaştırılması 33](#_Toc104929057)

[3.3.5. Sonuçların Değerlendirilmesi 34](#_Toc104929058)

BÖLÜM 4. [SONUÇLAR 35](#_Toc104929059)

[KAYNAKLAR 36](#_Toc104929060)

# SEMBOLLER LİSTESİ / SYMBOLS LIST

**w:** Ağırlık vektörü

**x:** Girdi vektörü

**b:** Sapma ()

Euler sayısı

# KISALTMALAR/ABBREVIATIONS

**SVM:** Support Vector Machine

**RNN:** Recurrent Neural Network

**kNN:** k-Nearest Neighbours

**XGBoost:** Extreme Gradient Boosting

**LR:** Linear Regression

**URL:** Uniform Resource Locator

**HTML:** HyperText Markup Language

**XML:** Extensible Markup Language

**CSV:** Comma Seperated Values

**LSTM:** Long Short-Term Memory

**MAE:** Mean Absolute Error

**MSE:** Mean Squared Error

**SQL:** Structured Query Language

**RegEx:** Regular Expression

**R2:**

# ŞEKİL LİSTESİ

Sayfa

Şekil 1.1. Web Kazıma İşleminin İş Akışı……………………….………………………..14

Şekil 1.2. Ayrıştırma İşlemi İş Akışı……………....…………………..…………………..15

Şekil 1.3. Araç Fiyat Bilgisi Ayrıştırma İşlemi.………………………….………………..16

Şekil 1.4. Örnek bir HTTP istek ve yanıt döngüsü..………………………………………..17

Şekil 1.5. BeautifulSoup ve sağladığı avantajlar …………………………………………..18

Şekil 1.6. Örnek bir CSV dosyası içeriği …………………………………………………..19

Şekil 1.7. Örnek bir Lineer Regresyon düzlemi …………………………..………………..22

Şekil 1.8. Örnek bir SVM düzlemi…………….………………………………….………..23

Şekil 1.9. Temsili Random Forest Diyagramı ……………………………………………..24

Şekil 1.10. Örnek bir Decision Tree diyagramı ..…………………………………………..25

Şekil 1.11. Sigmoid Fonksiyonu …………………………………………………………..26

Şekil 3.1. Algoritmaların R2 skorlarının karşılaştırılması ….……………………………...36

# TABLO LİSTESİ

Sayfa

Tablo 3.1. Algoritmalar ve R2 skorları…………………………………………………....36

# ÖZET

**WEB SCRAPING VE MAKİNE ÖĞRENMESİ İLE ARAÇ FİYAT TAHMİNİ**

Günümüzde internet üstünde bulunan verinin her geçen gün artması ile veri bilimi günümüzün önemli bilim alanlarından biri haline gelmiştir. Veri bilimi sayesinde başta anlamsız gibi gözüken verilerin işlenmesi sonucu anlamlı çıktılar elde edilebilmektedir. Bu işlenecek verilerin web üstünden elde edilmesi için ‘Web Scraping’ işlemi uygulanmaktadır. Bu işlem kapsamında kaynak websiteler tarafından istenilen veriler elde edilir. Elde edilen veriler çeşitli amaçlara göre değişebilecek şekilde farklı makine öğrenmesi algoritmaları ile işlenip anlamlı çıktı alma işlemi, günümüz dünyasında ticari anlamda büyük önem taşımaktadır. Veri bilimi uygulamaları ticari anlamda çok farklı sektörlerde uygulanmaktadır. Müşterilerin tercihlerini anlayabilme ve önerilerde bulunabilme gibi elektronik ticaretten dijital reklam uygulamalarına kadar geniş bir alanda makine öğrenme uygulamaları ile tahmin etme, sınıflandırma gibi problemler çözülmektedir.

Bu tez çalışmasının amacı, varsayılan olarak ‘Scraping’ işlemine izin vermeyen bir internet sitesinden, engellemeleri bypass ederek otomobil ilanı verileri çekme işlemine ve ardından bu verilerle, makine öğrenmesi yardımıyla bir aracın fiyatının piyasadaki değerini tahmin etmeye çalışmaktır. Bu amaç için geliştirilen programda kullanılan programlama dili Python olup, çalışma kapsamında çeşitli Python kütüphanelerinden faydalanılmıştır. Çalışma sonucunda elde edilen verilerle oluşturulan veriseti üstünde birden fazla makine öğrenmesi algoritması uygulanmış ve çıktılar değerlendirilmiştir.

**ABSTRACT**

**WEB SCRAPING AND CAR PRICE PREDICTION USING MACHINE LEARNING**

Data science has become one of the important science fields of today, with the increase of data available on the internet day by day. Thanks to data science, meaningful outputs can be obtained as a result of processing data that seems meaningless at first. In order to obtain the data to be processed over the web, the 'Web Scraping' process is applied. Within the scope of this process, the data requested by the source websites is obtained. The process of processing the obtained data with different machine learning algorithms, which may change according to various purposes, and obtaining meaningful output is of great commercial importance in today's world. Data science applications are applied in many different sectors commercially. Problems such as estimation and classification are solved by machine learning applications in a wide range from electronic commerce to digital advertising applications, such as understanding the preferences of customers and making suggestions.

The purpose of this thesis is to try to extract automobile advertisement data from a website that does not allow 'Scraping' by default, bypassing the blocks, and then to try to predict the market value of a vehicle's price with the help of machine learning with this data. The programming language used in the program developed for this purpose is Python, and various Python libraries were used within the scope of the study. More than one machine learning algorithm was applied on the dataset created with the data obtained as a result of the study and the outputs were evaluated.

## GİRİŞ

### Giriş

Günden güne internet üstündeki veri boyutu artmaktadır [1]. Artan veri boyutu ile Büyük Veri ve Veri Bilimi alanları gelişme göstermiştir. Genel olarak bu alanlar, anlamsız verilerin anlamlandırılabilmesi için çeşitli çalışmalar göstermektedir. Veri bilimi alanı; veri madenciliği, veri analizi, veri mühendisliği, veritabanı yönetim ve mimarisi, veri görselleştirmesi, makine öğrenmesi ve yapay zeka alanlarında çalışma göstermektedir. Veri analizi kısmında, market verisi analizi kısmı şirketlere önemli ticari gelirler sağlamaktadır. Bu alanda elde edilen market verileri analiz edilerek markete dair çıkarımlarda bulunulur ve bu doğrultuda market stratejileri ve planlamaları gerçekleştirilir. Bu adımlar sonucunda daha doğru ve detaylı planlama ve strateji neticesinde şirketler daha fazla ticari gelir elde edebilmektedir.

Verilerin analizi için veri bilimi dünyasında yoğun olarak yapay zeka çözümleri kullanılmaktadır. Yapay zekanın alt dalı olan makine öğrenmesi; verilerin sınıflandırılması, verilerin birbirleriyle ilişkisini ortaya çıkarmada kullanılan önemli bir alandır. Birçok süreçte, veriseti kullanılan algoritmada bir model olarak eğitilmekte ve test veriseti ile karşılaştırılarak çeşitli istatiksel parametrelerin değerine göre çalışmanın başarısı kritize edilmektedir.

Makine öğrenmesi algoritmalarının en bilindikleri Linear Regression, Logistic Regression, Decision Tree, Support Vector Machine (SVM), Naive-Bayes, K-Nearest Neighbours (KNN), Random Forest, Cross Validation gibi algoritmalar sayılabilir. Bu algoritmaların kullanımı, veriseti ve amaca göre değişkenlik göstermektedir.

Bu tez kapsamında, Türkiye’de en çok ziyaret edilen ilan sitesinden otomobil verilerinin elde edilmesi ve daha sonrasında bu verileri bir makine öğrenmesi algoritması yardımı ile işleyip daha sonrasında araç fiyat tahmini yapılması planlanılmıştır. Bu tez çalışmasında verisetinin elde edileceği internet sitesi, ‘Web Scraping’ işlemini varsayılan olarak yasaklamıştır. Bu çalışmadaki en önemli kilometre taşlarından birisi de izin verilmeyen bir web sitesinden veri kazıma işleminin gerçekleştirilmesidir. Bu işlem sonrasında elde edilen araba ilanı verilerinin verisetine dönüştürülmesi işlemi gerçekleştiriliyor olup, bu veriseti ile çeşitli makine öğrenmesi algoritma uygulamalarının gerçekleştirilip araç fiyat tahmini yapılabilmesi hedeflenmektedir.

#### Problemin tanımı

Otomobil şirketleri her geçen gün yeni model otomobillerini tanıtmaktadırlar. Bu durum, otomobil piyasası üzerinde fiyat takibi yapmayı zorlaştırmaktadır. Bu durumdan kaynaklanan maddi zararlar yaşanmaktadır. İlerleyen zamanla daha fazla otomobil markasının piyasaya çıkması, var olan otomobil markalarının yeni seri otomobillerini duyurmaları ve piyasadaki araç çeşitliliğinin ve araç sayısının artması bu durumu giderek daha da zorlaştırmaktadır. Bu durumda kullanılmak üzere piyasadaki araç ilanlarının analiziyle fiyat tahmini yapabilmek gereklidir.

#### Amaç ve hedef

Bu tez çalışmasında, sürekli çeşitliliği ve sayısı artan otomobil sektöründe, ikinci el veya sıfır bir otomobilin aracının gerçekçi fiyat tahminini gerçekleştirmek amacıyla bir “Web Scraper” ve fiyat tahminini gerçekleştirecek makine öğrenmesi programının gerçekleştirilmesi ve performans ölçümü amaçlanmaktadır.

#### Ana katkılar

Kazıma işlemine izin vermeyen siteye Python dilinde yazılan kodlarla istek gönderilmiştir. Sitenin yasaklama yapısı anlaşılmaya çalışıp, geliştirilen ek yöntemlerle başarılı bir erişim sağlanılmıştır. Yine Python dilinde kodlarla veriler, bir veriseti düzenine sokulmuştur. Belirlenen makine öğrenmesi algoritmalarıyla işlenen veriseti üstünden elde edilen sonuçlar, emsali örneklerine göre daha iyi sonuçlar vermiştir. Ayrıca uygulanılan makine öğrenmesi algoritmalarının performansları ve Türkiye piyasasında bir otomobilin fiyatını belirleyen etmenler arasında yük dağılımı yapılmış olup, olası stratejiler için anlamlı çıktı üretilme durumu araştırılmıştır.

#### Tez organizasyonu

Bu tez, dört ana bölümden oluşmuştur. “Giriş” bölümünde tez çalışması kapsamında genel bilgilere ve literatür çalışmalarına yer verilmiştir. İkinci bölüm olan “Materyal ve Yöntem” kısmında kullanılan verisetinden, verisetinin elde edilmesi için yapılan çalışmalardan, makine öğrenmesi ile ilgili bilgilerden ve bu tez çalışması kapsamında geliştirilen programdan bahsedilmiştir. Üçüncü bölüm kapsamı olarak “Bulgular ve Tartışma” kısmında uygulanan deneylerden, deneylerin ölçüm ve analizlerinden, performanslarından ve sonuçlarından bahsedilmiştir. Son kısım olan “Sonuçlar” bölümünde ise tez çalışması özetle sonuca bağlanmış ve ileride nasıl çalışmalar yapılabileceği hususunda fikirler beyan edilmiştir.

### Web Scraping Nedir?

Web Scraping, Türkçe kelime anlamıyla “Web Kazıma” anlamına gelmektedir. Web Scraping kapsamında yapılan işlem, web üstünde yer alan bir websitesi üstündeki verilerin otomatik olarak bir program/kod parçacığı yardımı ile elde edilmesidir.

Bu işlem için çeşitli programlama dillerinde yine çeşitli kütüphaneler kullanılabilmektedir. Bu işlem için en yaygın kullanılan programlama dillerinden biri Python olup, “Requests” ve “BeautifulSoup” kütüphanesi sıklıkla kullanılmaktadır. Genel çalışma mantığı ilgili kaynağa bir “HTTP Request” gönderilip, gelen “HTML Response”taki önemli veriyi ayrıştırmak (parsing) üzerine kuruludur.

Text

Description automatically generated with medium confidence

**Şekil 1.1.** Web Kazıma İşleminin İş Akışı

Bu bölümde görüldüğü üzere iş akışı Website üstünde Web Scraping işlerinin uygulanmasının ardından yapılandırılmış verinin elde edilmesini göstermektedir (Şekil 1.1.).

#### Web Parsing (ayrıştırmak) nedir?

Web üstünde “Parsing” işlemi bir web sayfasına gönderilmiş istek üzerine gelen HTML yanıtını “HTML Tag”lerine (etiketlerine) göre ayrıştırmaya denmektedir. Bu yöntem ile istenilen veri hangi HTML Tag’i altında tutuluyorsa direkt ona erişim sağlanır ve o veri elde edilir.

Şekil 1.2.’de görüldüğü üzere kullanıcının bilgisayarı üstünden websitesine gönderilen http isteği, websitesi tarafından yanıtlanır ve HTML yanıtı bilgisayara geri gönderilir. Alınan HTML yanıtı, ayrıştırma işlemine tabi tutulur ve HTML yanıtından ayrıştırılmış veri, CSV formatındaki dosyaya yeni satır olacak şekilde yazılır.

Diagram

Description automatically generated

**Şekil 1.2.** Ayrıştırma İşlemi İş Akışı

Bu bölümde örnek bir HTML dokümanı üstünden araç verisi bilgisinin ayrıştırılma işlemi görülmektedir (Şekil 1.3.).

Text

Description automatically generated

**Şekil 1.3.** Araç Fiyat Bilgisi Ayrıştırma İşlemi

#### HTTP isteği ve HTML yanıtı

HTTP kelimesi “Hypertext Transfer Protocol” kelimelerinin kısaltılmış halidir [2]. Bu protokol, istemciden veya tarayıcıdan gelen istekler ile web sunucularından dönen yanıtlar arasındaki yolu sağlayan protokoldür. HTML yanıtları, resimler, videolar gibi benzeri veriler TCP kullanılarak http vasıtası ile iletilir. Toplam yedi farklı istek metodu bulunmasına karşın bunlardan dört adedi yaygın kullanılır. Bunlar “GET”, “POST”, “PUT” ve “DELETE” metodlarıdır. “GET” metodu, kaynaktan veri almak için kullanılırken, “POST” metodu verileri kaynağa göndermek için kullanılmaktadır. “PUT” metodu ise gönderilmiş veriyi güncellemek için kullanılırken, “DELETE” metodu belirtilen veriyi silmek için kullanılmaktadır.

HTML kelimesi ise “Hypertext Markup Language” kelimelerinin kısaltılmış halidir. HTML, web üstünde websiteleri oluşturmak için kullanılmaktadır. Kendine özel bir yapısı olan bu dilden dönen yanıt, yani HTML yanıtı, sayfa içeriğinin HTML kodlarıdır.

Bu bölümde bir “client” ve sunucu arasında HTTP isteğinin gönderilmesi ve yanıtın alınması gösterilmiştir (Şekil 1.4.).

Shape, arrow

Description automatically generated

**Şekil 1.4.** Örnek bir HTTP istek ve yanıt döngüsü

#### Python nedir?

Python dili, nesne yönelimli, birimsel ve yorumlamalı yüksek seviyeli bir programlama dilidir [3]. Komut bitimini “indentation” ismi verilen girintilerle algılamaktadır. Python, günümüz dünyasında yapay zeka, makine öğrenmesi, veri kazıma ve scripting işlemleri için sıklıkla kullanılmaktadır. Güncel topluluğu büyük olup, kütüphane sayısı ve çeşitliliği oldukça fazladır. Bunlardan “sklearn”, “TensorFlow” kütüphaneleri yapay zeka ve makine öğrenmesi uygulamalarında sıklıkla kullanılırken, veri kazıma işlemleri için sıklıkla “BeautifulSoup4” ve “Requests” kütüphanesi kullanılmaktadır. Aynı zamanda büyük işlem gücü gerektiren işlemler için internet üstünde Cloud Tabanlı Computing çözümleri geliştirilmiştir. Bunlara Google’ın geliştirdiği Google Collaboratory servisi örnek verilebilir.

##### Google Colaboratory

Google Colaboratory veya diğer adıyla Google Colab, yapay zeka, derin öğrenme ve makine öğrenmesi projeleri üzerinde çalışanlar için tamamıyla bulut tabanlı, etkileşimli ve ortak çalışma sağlanabilen bir programlama ortamıdır [4]. Ücretsiz olarak kullanılabilir ve kurulum gerektirmez. Kullanıcıların Python kodlarını internet üstünde Colab sunucularında çalıştırmasını ve çıktılar alabilmesini sağlar.

##### BeautifulSoup4 Kütüphanesi

BeautifulSoup, Python’da bulunan ve web sayfalarından veri kazıma işlemlerini kolaylaştıran bir kütüphanedir [5]. Bir HTML veya XML dosyasının içeriğini verilen argümana göre kolaylıkla ayrıştırıp, içeriği çıktı olarak verebilir. Kullanım avantajları olarak, web sayfalarından rahatlıkla tüm başlıkları, linkleri, HTML etiketlerini ve HTML etiketleri içerisindeki text değerlerini ayrıştırabilir. Ayriyeten veri filtrelemesi, veri navigasyonu gibi artıları mevcuttur. Python’da sıklıkla kullanılan veri ayrıştırma kütüphaneleri olan “lxml” ve “html5lib” ile uyumlu çalışabilmesi sayesinde üstünde farklı ayrıştırma stratejileri uygulanılmasına müsait bir ortam sağlar.

Şekil 1.5.’te BeautifulSoup kütüphanesinin sağladığı avantajlar belirtilmektedir.

Graphical user interface, application

Description automatically generated

**Şekil 1.5.** BeautifulSoup ve sağladığı avantajlar

##### Requests Kütüphanesi

Requests kütüphanesi Python’da yazılmış Apache2 lisanslı bir HTTP kütüphanesidir. Requests kütüphanesi, kullanıcılara Python üstünden HTTP/1.1 istekleri gönderebilmelerini sağlar [6]. Kütüphane sayesinde isteklere başlıklar, form verileri, dosyalar ve parametreler eklenebilmektedir.

##### csv Kütüphanesi

CSV (Comma Seperated Values), tab lo ve veritabanlarında kullanılan en yaygın format biçimidir. Bu formatta farklı sütunlara ait veriler, virgül ile ayrılmaktadır. CSV formatı, RFC 4180’de standartları tanımlanmadan önce de uzun yıllar kullanılmıştır [7].

Python’da gömülü bulunan “csv” kütüphanesi, bu formatın dil içinde aktif olarak kullanılabilmesini sağlamaktadır. “csv” kütüphanesi sayesinde CSV formatındaki veriler, programın içerisine aktarılabilir veya dışarıya CSV formatında çıktı verilebilir. CSV formatındaki veriler üstünde kolaylıkla navigasyon yapılabilir veya bu veriler program içerisinde değiştirilip güncellenebilir.

Şekil 1.6.’da örnek bir CSV dosyası içeriği belirtilmektedir.

A close-up of a document

Description automatically generated with medium confidence

**Şekil 1.6.** Örnek bir CSV dosyası içeriği

##### Numpy Kütüphanesi

NumPy kütüphanesi, Python’da bilimsel araştırmalar için kullanılmakta olan bir kütüphanedir [8]. Kütüphane kapsamında kullanıcıya, diziler ve matrisler ile geniş bir çalışma alanı sunulmakta olup, Fourier dönüşümü gibi çeşitli matematiksel fonksiyonlar da gerçekleştirilebilir. NumPy sayesinde Python’da gömülü gelen dizilerden çok daha hızlı çalışan dizi değişkenleri oluşturulabilir ve kullanılabilir. Bu dizilerin, normal dizilerden çok daha hızlı olması sebebiyle genel olarak veri bilimi alanında sıklıkla kullanılmaktadır. Fakat NumPy dizilerinin, Python dizilerinin aksine homojen yapıda olması gerekmektedir. Bu da dizinin içerisindeki tüm elemanların aynı veri tipinden olmasını gerekli kılmaktadır.

##### Pandas Kütüphanesi

Pandas kütüphanesi, Python’da önceden toplanan verilerin analize uygun hale getirilmesi için kullanılan popüler bir kütüphanedir [9]. Kütüphane kapsamında veriler organize edilerek analizler daha kolay ve hızlı şekilde gerçekleştirilir. Genel olarak Excel’de veya SQL tablosunda yazılmış veriler, Pandas ile kolaylıkla analiz edilebilir. Pandas’ın NumPy’a benzer fonksiyonları bulunmakla birlikte, birbirlerinden ayrılan en önemli farkları NumPy üstünde yalnızca aynı veri tipinden çalışma sağlanabilirken, Pandas ile aynı anda farklı veri tipleri üstünde çalışılabilir. Pandas kapsamında en çok kullanılan veri yapıları, Series ve DataFrame’dir. Series veri yapısı, tek boyutlu bir veri yapısıyken, DataFrame veri yapısı iki boyutlu yani satır ve sütunlardan oluşan bir veri yapısıdır.

##### Seaborn Kütüphanesi

Seaborn kütüphanesi, Python’da açıklayıcı istatistiksel grafikler oluşturmaya yarayan bir kütüphanedir [10]. Matplotlib üzerine inşa edilmiş bir kütüphane olup; NumPy, Pandas gibi kütüphanelerin veri yapıları ile uyumlu çalışabilmektedir. Ayrıca verilerinizle oluşturacağınız istatiksel grafikler üstünde geniş bir özelleştirme seçeneği sunmaktadır.

##### Matplotlib Kütüphanesi

Matplotlib kütüphanesi Python’da bulunan ve veri görselleştirmeye yarayan bir kütüphanedir [11]. Matplotlib, sıklıkla iki boyutlu verilerin görselleştirilmesinde kullanılmaktadır. Kolay kullanımı ve popülerliği sebebi ile sıklıkla veri bilimi alanında verilerin görselleştirilmesinde kullanılmaktadır.

##### Scikit-learn Kütüphanesi

Scikit-learn kütüphanesi, Python’da veri bilimi ve makine öğrenmesi alanlarında sıklıkla kullanılan bir kütüphanedir [12]. Scikit-learn kapsamında birçok işlem gerçekleştirilebilmektedir. Bu işlemlere veri işleme, boyutsal küçülme, model seçimi, regresyon, sınıflandırma ve küme analizi gibi işlemler örnek verilebilir.

##### Statsmodels Kütüphanesi

statsmodel kütüphanesi, Python’da birçok farklı istatiksel model uygulanmasını sağlayan sınıflara ve fonksiyonlara sahiptir [13]. Ayrıca istatistiksel testler ve istatistiksel veri keşfi sağlayan bir kütüphanedir. Genel olarak kullanım alanı; veri analizi, veri bilimi ve istatistik dalıdır. NumPy ve SciPy’dan ayrılan en önemli noktası, daha gelişmiş istatistiksel test ve modelleme araçlarını bulundurmasıdır.

### Makine öğrenmesi

Makine öğrenmesi, çok büyük miktarlarda verinin elle işlenmesi ve analizinin mümkün olmadığı durumlar için geliştirilen yöntemlerdir. Hedef, geçmişteki verilerin kullanılarak işlenmesi ve bu doğrultuda gelecek için tahminde bulunma, anlamlı veri oluşturma işlemleri için kullanmaktır.

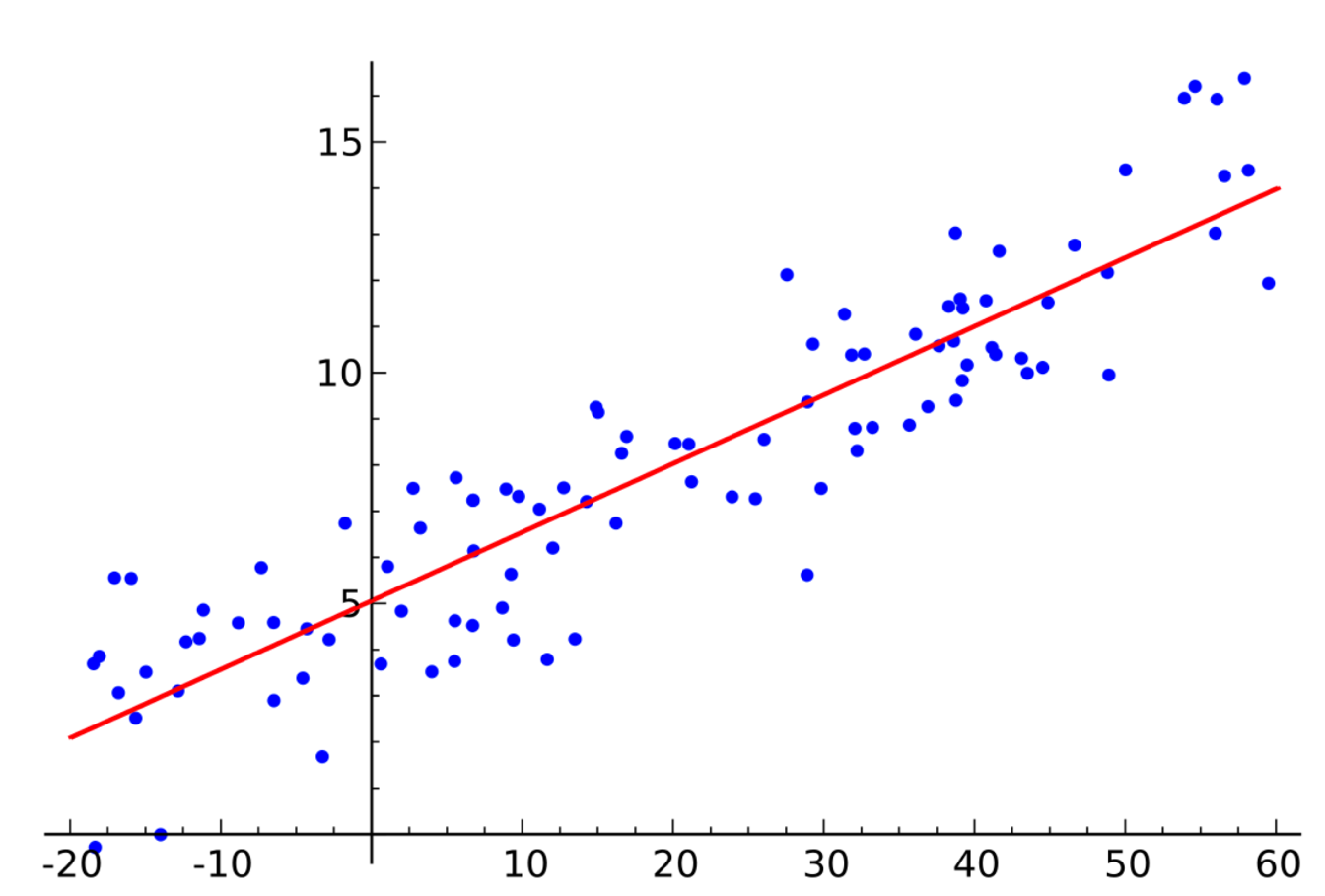
#### Makine öğrenmesi algoritmaları

Makine öğrenmesi uygulamaları için geliştirilmiş çeşitli algoritmalar bulunmaktadır. Bu algoritmalar; kullanım alanı, veriseti tipi ve amaca göre farklılıklar göstermektedir. Linear Regression (Lineer Regresyon), Support Vector Machines (SVM), Random Forest, Decision Tree gibi algoritmalar, sıklıkla kullanılmaktadır.

##### Linear Regression (Lineer Regresyon)

Lineer regresyon, bir dizi nokta için en optimal hiper düzlemi tespit etmek için kullanılmaktadır. Hiper düzlemin tespiti ile oluşturulacak yeni noktanın x-eksenindeki değerine göre, bu değerin hiper düzlemde kesiştiği noktanın y-eksenindeki değerinin esas alınması prensibi ile farazi bir noktanın herhangi bir eksendeki değerinin bilinmesi ile diğer eksendeki değerinin tahmininin yapılabilmesi söz konusudur.

Şekil 1.7.’de örnek bir lineer regresyon düzlemi görseli bulunmaktadır.

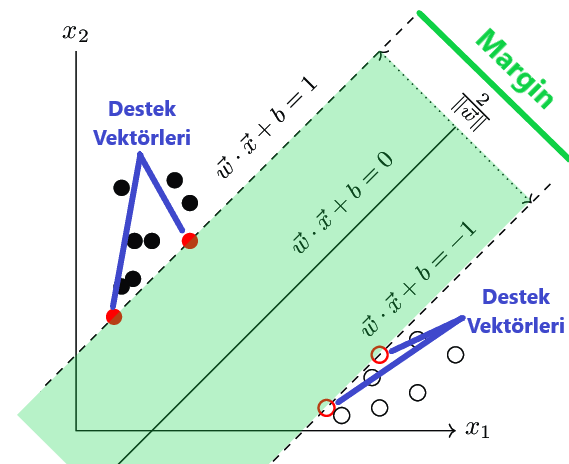


**Şekil 1.7.** Örnek bir Lineer Regresyon düzlemi

##### Support Vector Machines (SVM)

Support Vector Machines, Türkçe karşılığı olarak Destek Vektör Makineleri anlamına gelir. SVM, genellikle sınıflandırma senaryolarında kullanılan gözetimli öğrenme yöntemlerinden biridir. Genel olarak işleyişi, bir düzlem üzerine yerleştirilmiş noktaları ayırmak için bir doğru çizmektir. Bu doğru, düzlemdeki noktaları iki farklı gruba ayırır. Bu doğrunun ayırmış olduğu iki farklı gruptaki noktaların, çizilen doğruya maksimum uzaklıkta olması amaçlanmaktadır.

Bu bölümde örnek bir SVM düzlemi, Destek vektörleri ve marjini de içinde bulunduracak şekilde gösterilmiştir (Şekil 1.8.).



**Şekil 1.8.** Örnek bir SVM düzlemi

Şekil 1.8.’de beyaz ve siyahlar olacak şekilde iki farklı sınıf bulunmaktadır. Sınıflandırma ve tahmin problemlerinde asıl amaç, yeni gelecek verinin hangi sınıfta olacağına karar vermeye dayalıdır. Sınıflandırma işleminin gerçekleştirilmesi için iki sınıfı ayırmakta olan bir doğru çizilmektedir ve bu doğrunun ±1'i arasında kalan yeşil bölgeye “Margin” adı verilmektedir. Bu yeşil bölge ne kadar geniş olur ise sınıflar o kadar iyi ayrıştırılır.

A picture containing text, clock, antenna

Description automatically generatedMatematiksel olarak izahı denklem 1.1’de görülmektedir;

(1.1)

.: X vektörünün w üzerindeki izdüşümü)

Denklem 1.1’de görüldüğü üzere yeni bir değer için çıkan sonuç 0'dan küçükse, beyaz noktalara daha yakın olacaktır. Tam tersi, çıkan sonuç 0'a eşit veya büyükse, bu durumda siyah noktalara daha yakın olacaktır.

##### Random Forest

Random Forest, denetimli bir makine öğrenimi algoritmasıdır. Genel olarak sınıflandırma ve regresyon problemleri için kullanılmaktadır. Algoritma kapsamında farklı örnekler üzerinde karar ağaçları oluşturulur ve regresyon durumunda sınıflandırma ve ortalama için çoğunluk oyu kullanır.

Şekil 1.9.’da çoklu Decision Tree’lerden oluşan bir Random Forest diyagramı görülmektedir.

Diagram

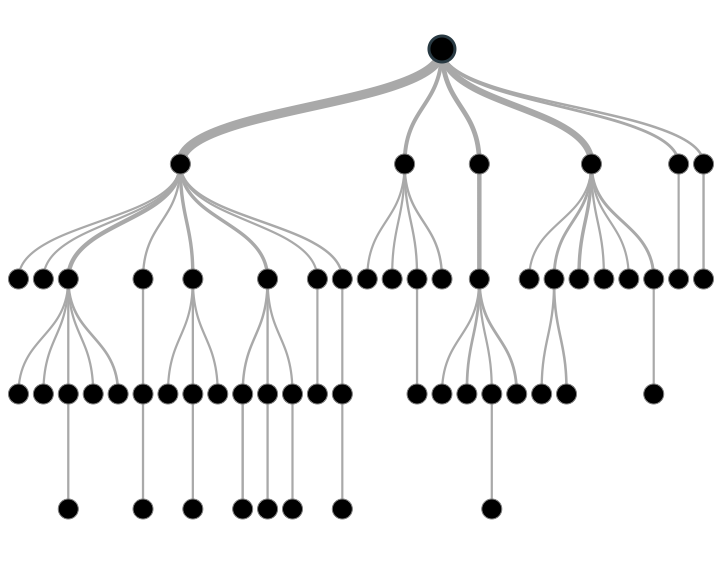
Description automatically generated

**Şekil 1.9.** Temsili Random Forest Diyagramı

##### Decision Trees

Decision Trees, Türkçe karşılığı ile “Karar Ağaçları”, amaç, sınıflama ve hususiyete yönelik karar düğümlerinden ve yaprak düğümlerinden oluşan ağaç yapısı düzeninde model oluşturan bir sınıflandırma metodudur. Karar ağaçlarının yapısı genel olarak programlama dillerinde mevcut olan “if-else” yapısına oldukça benzerlik göstermektedir. Metot kapsamında veriler, belirli kriterlere göre birbirinden ayrılarak nihai anlamlı sonuca erişilmeye çalışılır ve veri doğru kategoride sınıflandırılmaya çalışılır.

Şekil 1.10.’da örnek bir Decision Tree diyagramı bulunmaktadır.



**Şekil 1.10.** Örnek bir Decision Tree diyagramı

### Sınıflandırma

Makine öğrenmesinin uygulama alanlarından biri olan sınıflandırma, bir veri kümesinin birbirinden farklı ve önceden belirlenmiş sınıflardan birine atanmasıdır. Sınıflandırma işleminde kullanılan algoritmalar, verilen eğitim kümesinden hangi veriyi hangi sınıfa atayacağını öğrenmektedir. Öğrenme işleminden sonra ise test verilerini ilgili sınıflara atamaya çalışır.

Sınıflandırma fonksiyonu olarak sıklıkla kullanılan Sigmoid Fonksiyonu şu şekildedir (Denklem 1.2):

(1.2)

( Girdi,Euler sayısı )

Yukarıda belirtilen fonksiyon, gerçek bir sayıyı (0,1) aralığına eşler. Bu durum, rastgele bir değere sahip bir fonksiyonun sınıflandırma için daha uygun bir fonksiyona dönüştürülmesini sağlar.



**Şekil 1.11.** Sigmoid Fonksiyonu

### Literatür Çalışmaları

Web Scraping ve fiyat tahminlemesine yönelik makine öğrenimi uygulamalarına ilişkin birçok çalışma mevcuttur. Bu bölümde, bu alanda daha önceden literatüre kazandırılmış çalışmaların özetleri yer almaktadır.

Mehar, Deeksha, Vinay & Arun [14], yapmış oldukları çalışmada stok fiyatlarının tahminlemesi üzerine yapay sinir ağları kullanmışlardır. Yaptıkları çalışmada kurmuş oldukları yapay sinir ağları mimarisinde üç farklı katman kullanmış olup, bu katmanlar sonucunda elde edilen çıktı Random Forest metoduna tabi tutulmuştur. Kurmuş oldukları bu mekanizmayı beş farklı şirketin stok verileri üstünde uygulamış olup, alınan sonuçlar paylaşılmıştır.

Mahla, Gholamreza & Jamshid [15], yapmış oldukları çalışmada stok fiyatı tahmini gerçekleştirmeye çalışmışlardır. Kurmuş oldukları yapıda Yinelemeli Sinir Ağları (Recurrent Neural Networks - RNN), SVM ve Random Forest algoritmalarını kullanmışlardır. Çalışmada yalnızca makine öğrenmesi kapsamında sıklıkla kullanılan SVM ve Random Forest kullanılmamış olup, Makine Öğrenmesinin alt dallarından biri olarak kabul edilen derin öğrenme metodları da uygulanarak anlamlı çıktıların elde edilmesi amaçlanmıştır.

Zheshi, Chunhong & Wenjun [16], yapmış oldukları çalışmada günümüzde popülerliğini koruyan kripto para birimlerinden biri olan Bitcoin hakkında fiyat tahminlemesi üzerine çalışmışlardır. Çalışmalarında kullandıkları metotları iki kategoriye ayırıp, bu kategorileri “İstatistiksel Metotlar” ve “Makine Öğrenmesi Modelleri” şeklinde isimlendirmişlerdir. “İstatistiksel Metotlar” içerisinde iki farklı metot kullanılmış olup, bunlar “Logistic Regression” ve “Linear Discriminant Analysis”tir. “Makine Öğrenmesi Modelleri” içerisinde ise beş farklı metod kullanılmış olup, bunlar “Random Forest”, “XGBoost”, “Quadratic Discriminant Analysis”, “Support Vector Machine” ve “Long Short-term Memory”dir. Yapmış oldukları araştırma neticesinde günlük Bitcoin fiyat değişim değerleri üstünde uygulanan metotlar arasında en yüksek “Accuracy” ve “Precision” değerinin “Logistic Regression” metodunda elde edildiği görülmüştür.

Danh [17] tarafından yapılan çalışmada Avusturya’nın Melbourne şehrindeki ev fiyatları üstüne fiyat tahminlemesi üstüne çalışılmıştır. Çalışmada kullanılan veri setinde Melbourne kentinde 2016-2018 arasında satılan 34.857 adet evin bilgileri yer almaktadır. Kullanılan metotlar “Stepwise”, “Principal Component Analysis”, “Polynomial Regression”, “Regression Trees”, “Neural Network” ve “Support Vector Machine” olmak üzere altı farklı ana metottur. Elde edilen sonuçlarda, en iyi değerlendirme oranına sahip metot “Neural Network” olmuş, ikinci en iyi orana sahip metot ise Doğrusal Regresyon olmuştur.

Sameerchand [18] tarafından yapılmış olan çalışmada ise ikinci el araçların fiyat tahmini makine öğrenmesi teknikleri ile gerçekleştirilmeye çalışılmıştır. 400’den fazla Toyota marka araç verisi kullanılarak oluşturulan veri seti üstünde Çoklu Doğrusal Regresyon, K-Nearest Neighbours, Decision Trees, Naïve Bayes olmak üzere 4 farklı metot uygulanmıştır. Alınan sonuçlarda kNN ve Doğrusal Regresyon metotları, diğer metotlara göre daha başarılı sonuç göstermiştir.

Nitis ve diğerlerinin [19] yapmış olduğu çalışmada, 371.528 araç verisi toplanmış olup oluşturulan veri seti kullanılarak ikinci el araç fiyatı tahminlemesi gerçekleştirilmeye çalışılmıştır. Metodoloji olarak regresyon metotları tercih edilmiş olup, bunlar Çoklu Doğrusal Regresyon ve Random Forest Regresyonu’udur. Modellerin uygulanımı neticesinde alınan sonuçlar, Çoklu Doğrusal Regresyon’da MSE (Mean Absolute Error) değeri 0.55 olarak hesaplanmışken; Random Forest Regresyonu neticesinde alınan MSE değeri 0.35’tir.

Prashant ve diğerlerinin [20] yapmış olduğu çalışmada beş farklı algoritmanın implementasyonu neticesinde ikinci el araçların fiyat tahmini gerçekleştirilmeye çalışılmıştır. Bu algoritmalar kNN, Random Forest, Doğrusal Regresyon, XG Boost ve Decision Trees’tir. Alınan sonuçlar neticesinde bu beş farklı algoritmadan en yüksek Accuracy değerini veren algoritma Random Forest olmuş olup, onu ikinci sırada XG Boost takip etmiştir.

Mariana [21] tarafından gerçekleştirilmiş olan çalışmada, araç kiralama için fiyat tahminlemesi gerçekleştirilmesi hedeflenmiştir. Veri seti olarak 124,386 veri içeren bir veri seti kullanılmış olup, metot olarak Doğrusal Regresyon ve SVM kullanılmıştır.

Quang ve diğerlerinin [22] yapmış olduğu çalışmada geliştirilmiş makine öğrenmesi teknikleri ile ev fiyatlarının tahminlemesi üzerine çalışılmıştır. 26 farklı değişken içeren 300.000 satırlık ev verisi, veriseti olarak kullanılmıştır. Algoritma olarak Random Forest ve Extreme Gradient Boosting (XGBoost), Light Gradient Boosting Machine (LightGBM), Hybrid Regression ve Stacked Generalization kullanılmıştır. Alınan sonuçlar neticesinde eğitim setinde en iyi sonucu alan algoritmanın Random Forest, test setinde alınan en iyi sonucun ise Stacked Generalization Regression algoritmasında alındığı görülmüştür.

Yili ve Yiting’in yapmış olduğu çalışmada [23], petrol keşifleri ve petrol endüstrisinin geliştirilmesinde kullanılabilecek bir Web Scraper geliştirmeyi amaçlamışlardır. Araştırma kapsamında, araştırmayı gerçekledikleri programlama dili Python olarak belirlenmiştir. Web Scraper geliştirme amaçlı Python programlama dili üstünde veri kazıma kütüphanelerinden biri olan Scrapy kütüphanesini kullanmışlardır. Elde edilen veriler üstünde K-Means algoritması kullanılarak verilerin kümelenmesi hedeflenmiştir.

Sirisuriya [24] tarafından yapılmış olan çalışmada, Web Scraping teknolojilerinin yapısından, günümüz dünyasındaki kullanım alanlarından ve web kazıma tekniklerinden bahsetmiştir. Ayriyeten veri ayrıştırma ve veri kazıma amaçlı kullanılan araç ve yazılımlardan bahsedip, bu araç ve yazılımları kendi aralarında karşılaştırılmasını gerçekleştirmiştir.

Thivaharan ve diğerlerinin [25] yapmış olduğu çalışmada, Python’da kullanılan web kazıma kütüphanelerinin performans karşılaştırılmaları gözlemlenmiştir. Web kazıma kütüphaneleri olarak RegEx, BeautifulSoup ve Lxml kütüphaneleri seçilmiştir. Bu üç kütüphanenin aynı veri üstünde veri ayrıştırma performansları ölçülmüştür. Ölçümler sonucunda RegEx kütüphanesinin, BeautifulSoup ve Lxml kütüphaneleri ile olan kıyaslamasında daha hızlı yanıt verebildiği görülmüştür.

Pratiba ve diğerlerinin [26] yapmış olduğu çalışmada, Python’da web kazıma teknikleri ile Google Scholar’dan veri kazımını hedeflemişlerdir. Yapmış oldukları çalışmada kütühaneler olarak pymysql, csv, requests, BeautifulSoup, pandas, and tkinter kullanmışlardır. Araştırma kapsamında grafik arayüzlü bir program tasarlanmış olup, Scholar üstünden elde edilen veriler, CSV ve SQL tablosu formatında çıktılarla kullanıcıya sağlanmıştır.

Fatmasari [27] ve diğerlerinin çalışmasında, web kazıma teknikleri ile Güney Sumatra bölgesinin hava durumu bilgileri elde edilmeye çalışılmıştır. Araştırma kapsamında kullanılan programlama dili Python’dır. Web kazıma için kullanılan kütüphaneler, BeautifulSoup ve Requests kütüphaneleridir.

Manik ve diğerlerinin [28] yapmış olduğu çalışmada, makine öğrenme teknikleri ile mahsul fiyatlarının tahminlemesi üzerine çalışılmıştır. Araştırma kapsamında yaygın kullanılan makine öğrenme algoritmaları olan Linear Regression, Decision Tree, Support Vector Machine, Naive Bayes, k-Nearest Neighbours, K-Means, Random Forest, Dimensionality Reduction Algorithms, Gradient Boosting Machine, XGBoost, LightGBM ve CatBoost hakkında bilgi verilmiştir. Ardından var olan veriler, Linear Regression ve Decision Tree algoritmalarına tabi tutulup, sonuçlar paylaşılmıştır.

Abhishek ve diğerlerinin [29] yapmış olduğu araştırmada, makine öğrenmesi ile araç fiyatı tahmini gerçekleştirilmesi amaçlanmıştır. Araştırma kapsamında kullanılan programlama dili Python olup, kullanılan kütüphaneler Pandas, NumPy, Matplotlib, Seaborn ve Scikit-Learn’dür. Veri seti internet üzerinden alınmış olup, Random Forest ve ExtraTree Regression isimli algoritma ile uygulamalar gerçekleştirilmiştir.

Shanu ve diğerlerinin [30] yapmış olduğu çalışmada, toplumdaki bireylerin sağlık giderleri tutarları makine öğrenmesi teknikleri ile tahmin edilmeye çalışılmıştır. Çalışma kapsamında iki farklı veri seti kullanılmıştır. Araştırmada kullanılan makine öğrenme metotları olarak Linear Regression, Regression Tree, Random Forest ve M5 Model Tree kullanılmıştır. Bu algoritmalar, her iki veri setine de uygulanmış olup, algoritmalardan alınan sonuçlar birbirleri ile karşılaştırılmıştır.

## MATERYAL VE YÖNTEM

### Veriseti elde edinimi

Bu çalışma kapsamında hazır veriseti kullanılmamış olup, kullanılan veriseti araştırmacılar tarafından “Web Kazıma” metodu ile elde edilmiştir. Kazım yapılan site, Türkiye’de otomobil ilanlarının paylaşıldığı en yaygın ve popüler sitedir. Önce websitesinden kazım yapılmış olup, kazım sonrası elde edilen veriler üstünde bir verisetine uyacak şekilde düzenlemelerde bulunulmuştur.

#### Websitesinden veri kazıma süreci

Öncelikle, araştırma kapsamında hedef olarak alınan websitesi web kazıma işlemlerine varsayılan olarak blokaj koymaktadır. Websiteleri, içeriğinin dışarıdan elde edilmesi ve kazıma süreçleri için olan izinlerini sayfa dizinlerinde bulunan “robots.txt” dosyasında tutarlar. Bu çalışmada veri kazımı için belirlenen site, dışarıdan gelen robot istekleri otomatik olarak engellemektedir.

Sitenin robot cihazlardan gelen istekleri otomatik olarak tespiti ve yasaklamasından mütevellit, öncelikli amaç websitesine stabil ve düzenli olarak içerik kazımayı sağlayacak altyapıyı kurmaktır. Bu altyapıyı kurmak için site özelinde sitenin yasaklama mekanizmasını keşfi ve analizi kritik rol oynamaktadır.

Websitesi, Python kodundan gönderilen isteklerin robot istek olduğunu direkt tespit edebilmiş ve erişimi engellemiştir. Buna çözüm olarak istek bazında insan davranışı simüle edilmiş, gönderilen istek dosyasının başlığına “User-Agent” ve sitenin kendisinin güvenlik amaçlı kullandığı istek başlık bilgileri de eklenmiştir. Bu konfigürasyonla gönderilen istekler, website tarafından istek bazında robot istek olduğu tespit edilememiş, çok istek gönderiminde aynı network üstünden gönderim sağlanması sebebi ile network tarafında tespit edilmiştir. Bu problemin de aşılabilmesi adına istek gönderilen ağ üstünde “Proxy Rotating” işlemi gerçekleştirilmiştir. Bu işlem kapsamında Python kodu üstünden isteğin gönderildiği cihazın proxy’si belirli istek sayısı aralıklarında değiştirilmiş, bu sebeple web sayfasına aynı ağ üstünden yüksek sayıda istek gönderilmemesi hedeflenmiştir. Bu yöntem ile web sayfası aynı ağ üzerinde çoklu istek gönderimi göremediği için engelleme koyamamış, web sayfasından başarılı ve stabil bir surette veri çekme işlemleri gerçekleştirilmiştir.

#### Proxy Rotating işlemi

Websitesinin engel mekanizması kapsamında aynı ağ üzerinden çoklu istek gönderimini engellediği tespit edilmiş olup; bu engel, düzenli aralıklarla proxy değiştirmeyle aşılmıştır. Python kodu içerisinde yapılan implementasyonlarla, isteğin gönderildiği ağ konfigüre edilmiş olup, aynı proxy üstünden belirli bir istek gönderilimi sayısına ulaşıldığında farklı bir proxy kullanımına başlanmış olup, istekler o şekilde gönderilmeye devam edilmiştir.

#### Kazımı yapılan verinin verisetine dönüştürülme süreci

Çekilen veri kapsamında BMW marka araçların “3 Serisi” model araçları hedef alınmıştır. Toplamda 904 adet otomobil ilanı verisi kazılmış olup, otomobil ilanlarının otomobile dair bilgileri taşıdığı website alanları hedef alınmıştır. Bu kapsamda, her bir otomobilin modeli, yılı, yakıt tipi (dizel, benzin, benzin+lpg), vites tipi (otomatik, yarı otomatik, manuel), kilometresi, beygir gücü, motor hacmi ve fiyat bilgileri elde edilmiştir.

Çekilen veriler, sütunlar otomobilin modeli, yılı, yakıt tipi, vites tipi, kilometresi, beygir gücü, motor hacmi ve fiyat bilgisi olacak şekilde, her bir satır da bir yeni otomobil olacak şekilde bir “.csv” uzantılı dosyada depolanmıştır. Araç fiyatı, kilometresi gibi sütunlarda bulunan “123.456 TL” gibi değerlerde noktalar, boşluklar ve birim isimleri silinerek düzeltilmiş, üstlerinde matematiksel ve mantıksal işlemler yapmaya uygun hale getirilmiştir.

### Verisetinin işlenmesi

Bu bölümde otomobillerin fiyat tahmini için kullanılan makine öğrenmesi algoritmaları ve istatiksel metotların genel yapılarından ve projeye implementasyonları hakkında genel bilgiler verilmektedir.

#### OLS (Ordinary Least Squares, Sıradan En Küçük Kareler Yöntemi) uygulanması

“Sıradan En Küçük Kareler” yöntemi, birbirine bağlı olarak değişen iki fiziksel büyüklük arasındaki ilişkiyi olabildikçe gerçeğe uygun şekilde bir denklem olarak yazmak için kullanılan bir regresyon yöntemidir. Bu yöntem sayesinde, gözlemi yapılan yanıtlar arasındaki değişkenlerin karelerinin toplamını en aza indirgemek hedefi ile lineer regresyon modelinde bilinmeyen parametreler tahmin edilir.

Proje kapsamında verisetine OLS yöntemi uygulanmış olup, alınan çıktı üstünden verilerin “P değeri” hesaplanmıştır.

#### P Değeri uygulanması

P değeri, istatistikte kullanılan ve gözlemlenen bir örneklem sonuçlarının ne kadar aşırı olduğunu ölçmek için kullanılan bir yöntemdir.

“Sıradan En Küçük Kareler” yönteminin uygulanmış olduğu verisetine “P değeri” ölçümleri gerçekleştirilmiş olup, aykırı değerlerin makine öğrenimine negatif etkilerinden kaçınılmıştır.

#### Model Eğitimi

Model eğitimi için “P Değeri” uygulamasından çıkmış olan veri farklı makine öğrenmesi algoritmaları ile eğitilmiş olup, ayriyeten “Cross Validation” metodu ile veriseti üstünde performans testi işlemleri gerçekleştirilmiştir.

Proje kapsamında makine öğrenimi algoritmaları olarak Lineer Regresyon, Random Forest ve Decision Trees olmak üzere farklı makine öğrenimi algoritması kullanılmıştır.

##### Cross Validation

Cross Validation (Çapraz doğrulama) yöntemi, makine öğrenmesi modelinin henüz görmediği veriler üstündeki performansını objektif ve doğru bir şekilde değerlendirmek amacıyla kullanılan bir yeniden örnekleme yöntemidir.

## BULGULAR VE TARTIŞMA

Bu bölümde, Web Scraping ve makine öğrenmesi işlemleri için kullanılan çalışma ortamının özelliklerinden, araştırmanın değerlendirme ölçütlerinden, hazırlanan verisetlerinin üzerinde belirtilen makine öğrenmesi algoritmalarının test edilmesinden, elde edilen sonuçlardan ve modelin başarı durumundan bahsedilmektedir. Araştırma kapsamında kullanılmış olan Doğrusal Regresyon, Random Forest ve Decision Tree algoritmalarının sonuçları teker teker incelenmiş olup, sonrasında bu algoritmaların kendi aralarındaki karşılaştırılmaları, belirlenmiş performans metriğine göre karşılaştırılmıştır.

### Deney Ortamı

Deneyler için kullanılan cihaz MacOS Big Sur işletim sistemine sahip, Intel Core i5 2.7 Ghz işlemci, 8GB RAM, 128GB SSD özelliklerini içeren 2015 model Macbook Pro bilgisayardır. Geliştirme ortamında IDE olarak PyCharm kullanılmış olup, programlama dili olarak Python’ın 3.7 versiyonu tercih edilmiştir. Programın koşturulmasında Google’ın Cloud Computing çözümü olan Google Colaboratory kullanılmıştır. Web Scraping işlemleri için Python ve BeautifulSoup4, Requests, NumPy ve Pandas kullanılmış olup, makine öğrenimi misyonları için yine Python programlama dilinde NumPy, Pandas, SKLearn, Seaborn, matplotlib kullanılmıştır.

### Performans Metrikleri

Modelin performans ölçümleri için verilerin regresyon hattına yakınlığının ölçümü olarak algoritmaların R2 skorları hesaplanarak esas alınmıştır. R2 skorunun hesaplanma formülü (3.1) numaralı denklemde belirtilmiştir.

(3.1)

### Deney Sonuçları

Çalışma kapsamında Web Scraping yöntemi ile web üzerinden kazılan verilerle oluşturulan veriseti üstünde uygulanmış olan Random Forest, Decision Tree ve Doğrusal Regresyon yöntemlerinin performans ölçümleri elde edilmiştir. Performans metriki olarak algoritmaların R2 skorlarının ölçümleri gerçekleştirilmiş olup, birbirleri ile performansları karşılaştırılmıştır.

#### Doğrusal Regresyon Sonuçları

Çalışma kapsamında Doğrusal Regresyona tutulan veriseti üstünde performans ölçümleri gerçekleştirilmiştir. Sonuçlara göre Doğrusal Regresyon metodu kapsamında alınmış olan R2 skoru 0.9208 olarak ölçülmüştür.

#### Decision Tree Sonuçları

Çalışma kapsamında Decision Tree metoduna tabi tutulan veriseti üstünde performans ölçümleri gerçekleştirilmiştir. Sonuçlara göre Decision Tree metodu kapsamında alınmış olan R2 skoru 0.9253 olarak ölçülmüştür.

#### Random Forest Sonuçları

Çalışma kapsamında Random Forest metoduna tabi tutulan veriseti üstünde performans ölçümleri gerçekleştirilmiştir. Sonuçlara göre Random Forest metodu kapsamında alınmış olan R2 skoru 0.9525 olarak ölçülmüştür.

#### Sonuçların Karşılaştırılması

Araştırma bulgularına göre veriseti üstünde kullanılan üç farklı algoritmanın R2 skorları hesaplanmış olup, bu üç metot arasında R2 skoru açısından en son sırada 0.9208 skorla Doğrusal Regresyon bulunuyor olup, onun sonrasında 0.9253 skorla Decision Tree metodu gelmektedir. İlk sırada ise Random Forest gelmekte olup, bu metot kapsamında alınmış olan skor, 0.9525 olarak hesaplanmıştır. Tablo 3.1.’de kullanılan algoritmaların R2 skorları belirtilmiştir. Şekil 3.1.’de de kullanılan algoritmaların R2 skorlarının yanyana olduğu bir grafik gösterilmiştir.

|  |  |
| --- | --- |
|  | **R2 Skoru** |
| **Linear Regression** | 0,920833949884016 |
| **Decision Tree** | 0,925357769970486 |
| **Random Forest** | 0,952597912661189 |

**Tablo 3.1.** Algoritmalar ve R2 skorları

A picture containing chart

Description automatically generated

**Şekil 3.1.** Algoritmaların R2 skorlarının karşılaştırılması

#### Sonuçların Değerlendirilmesi

Performans ölçümü sonuçları olarak üç metotta da yüksek R2 skoru elde edilebilmesine karşın, birbirleri arasında ufak farklılıklar gözlemlenmiştir. Performans sıralamasında üçüncü sırada gelen Doğrusal Regresyon ve ikinci sırada olan Decision Tree metotlarının R2 skorları arasında 0.0045 skor farkı bulunmaktadır. Yine ikinci sırada olan Decision Tree metodu ve birinci sıradaki Random Forest metodu arasında 0.0272 R2 skoru farkı bulunmaktadır. Kullanılmış olan bu üç metotta da alınmış olan skorlar yüksek olup, bu metotlar sayesinde yüksek doğrulukla araç fiyatı tahmini yapmaya elverişli ortam sağlanmıştır.

Ayriyeten yapılmış olan araştırmanın sonuçları kapsamında Türkiye’deki aynı marka aynı model ikinci el araç piyasasındaki en önemli fiyat indikatörünün aracın yıl değerinin olduğu tespit edilmiş olup, ikinci en önemli indikatörün aracın kilometre sayacı olduğu görülmüştür. Araçlardaki fiyat değişimleri, genel olarak aracın yıl değerinin günümüze yakınlığı arttıkça fiyatının artıyor olduğu görülmüş olup; aracın kilometre sayacında ise aracın yapmış olduğu kilometre ne kadar fazlaysa aracın ikinci el değerini o kadar kaybettiği görülmüştür.

# SONUÇLAR

Bu tez çalışmasında, veri kazımına izin vermeyen bir websitesinden veri çekme yöntemleri araştırılmış olup, websitelerinin veri kazımına olan engelleme mekanizmalarının keşif ve analizi gerçekleştirilmiştir. Websitelerinin koymuş olduğu bu engelleme mekanizmalarının aşılması ve etkisiz hale getirilmesi için alternatif yöntemler geliştirilmiş olup, websitelerinin kendilerine gelen ağ trafiğini algılaması ve kendi standartlarına göre anomali tespiti çalışmaları hakkında analizler yapılmış olup, bu websitelere gönderilebilecek olan robot isteklerin nasıl yakalanamayacağı hakkında çalışmalarda bulunulmuştur. Yapılmış olan çalışmalar neticesinde en son çalışan yöntemde, genel mantık olarak gönderilen robot isteğin, gerçek kullanıcı tarafından gönderilen isteğe benzetilmesi esas alınmıştır. “User-Agent” gibi veya websitelerin kendine has yetkilendirme yapan istek başlıkları parametreleri benzetilerek hazırlanan istekler, Türkiye lokasyonlu Proxy (vekil) sunucular tarafından istek sayıları belirli spesifik aralıklara bölünerek farklı farklı vekil sunucular vasıtasıyla gönderilmiş ve dönen yanıtlar üstünde ayrıştırma işlemi yapılarak websitesindeki ilgili HTML konumlarından gerekli bilgiler, düzenli bir şekilde depolanmıştır. Düzenli şekilde depolanmış olan bu veriler, tek bir dosya altında toplanırken; toplanan dosya üstünde veriseti hazırlama işlemleri başlatılmıştır. Veriseti hazırlandıktan sonra, bu elde edilen veriseti üstünde Doğrusal Regresyon, Decision Tree ve Random Forest olacak şekilde üç farklı makine öğrenmesi algoritması uygulanmış olup; bu algoritmaların performans ölçümleri R2 skoru metriğine göre gerçekleştirilmiştir. Algoritmaların performans ölçümleri kapsamında elde edilmiş olan R2 skorları, kendi aralarında karşılaştırılmış olup; en yüksek performansı ve en düşük performansı veren algoritmalar belirlenmiştir. Ayriyeten, Türkiye’deki ikinci el araç piyasasındaki en büyük fiyat indikatörünün aracın yıl değerinin olduğunun, ondan sonra da kilometre sayacı değeri olduğunun analizi yapılmış olup, ikinci el bir aracın fiyatını etkileyen hususlar hakkında çıkarımlarda bulunulmuştur.

Bu çalışmanın devamı olarak gelecekte, yalnızca araç fiyatı tahmini değil emlak sektöründeki konutların fiyatlarının tahmini de gerçekleştirilebilir olup; gerçek zamanlı veri kazıma işlemi ile kendi başına öğrenen bir yapı oluşturulması hedeflenmektedir. Kullanılan metotlar çoğaltılması, literatatürde sıklıkla kullanılan diğer farklı metotlara yer verilmesi planlanmaktadır. Ayriyeten gelecek çalışmalarda yalnızca makine öğrenmesi değil, derin öğrenme alanında yapay sinir ağları çeşitleri olan Recurrent Neural Network, Modular Neural Network, Feedforward Neural Network ve Kohonen Self Organizing Neural Network gibi yapay sinir ağlarından uygun olabilecek olanların seçilip projeye implementasyonu ve diğer yöntemlerle karşılaştırılması hedeflenmektedir.

# KAYNAKLAR

[1] E. Aktan, B. Yönetimi, H. Makaleler, and M. Bilgisi, “Büyük veri: Uygulama alanları, analitiği ve güvenlik boyutu,” *dergipark.org.tr*, Accessed: May 29, 2022. [Online]. Available: <https://dergipark.org.tr/en/pub/by/article/403010>

[2] <https://www.w3.org/Protocols/> (Erişim tarihi: 07.06.2022)

[3] <https://www.python.org/> (Erişim tarihi: 07.06.2022)

[4] <https://colab.research.google.com/> (Erişim tarihi: 07.06.2022)

[5] <https://pypi.org/project/beautifulsoup4/> (Erişim tarihi: 07.06.2022)

[6] <https://pypi.org/project/requests/> (Erişim tarihi: 07.06.2022)

[7] <https://datatracker.ietf.org/doc/html/rfc4180> (Erişim tarihi: 07.06.2022)

[8] <https://numpy.org/> (Erişim tarihi: 07.06.2022)

[9] <https://pandas.pydata.org/> (Erişim tarihi: 07.06.2022)

[10] <https://seaborn.pydata.org/> (Erişim tarihi: 07.06.2022)

[11] <https://matplotlib.org/> (Erişim tarihi: 07.06.2022)

[12] <https://scikit-learn.org/stable/> (Erişim tarihi: 07.06.2022)

[13] <https://www.statsmodels.org/stable/index.html> (Erişim tarihi: 07.06.2022)

[14] M. Vijh, D. Chandola, V. A. Tikkiwal, and A. Kumar, “Stock Closing Price Prediction using Machine Learning Techniques,” *Procedia Computer Science*, vol. 167, pp. 599–606, Jan. 2020, doi: 10.1016/J.PROCS.2020.03.326.

[15] M. Nikou, G. Mansourfar, and J. Bagherzadeh, “Stock price prediction using DEEP learning algorithm and its comparison with machine learning algorithms,” *Intelligent Systems in Accounting, Finance and Management*, vol. 26, no. 4, pp. 164–174, Oct. 2019, doi: 10.1002/ISAF.1459.

[16] Z. Chen, C. Li, and W. Sun, “Bitcoin price prediction using machine learning: An approach to sample dimension engineering,” *Journal of Computational and Applied Mathematics*, vol. 365, Feb. 2020, doi: 10.1016/J.CAM.2019.112395.

[17] T. D. Phan, “Housing price prediction using machine learning algorithms: The case of Melbourne city, Australia,” *Proceedings - International Conference on Machine Learning and Data Engineering, iCMLDE 2018*, pp. 8–13, Jan. 2019, doi: 10.1109/ICMLDE.2018.00017.

[18] S. Pudaruth, “Predicting the Price of Used Cars using Machine Learning Techniques Predict ing t he Price of Second-hand Cars using Art ificial Neural Net works Sameerchand Pudarut h DBST 667 Mult ilayer Percept ron Exercise Yelena Byt enskaya T he 6 t h Int ernat ional Conference on Cyber and IT Service Management (CIT SM 2018) Implement at io… fit riana harahap Predicting the Price of Used Cars using Machine Learning Techniques,” *International Journal of Information & Computation Technology*, vol. 4, no. 7, pp. 753–764, 2014, Accessed: May 29, 2022. [Online]. Available: <http://www.irphouse.com>

[19] N. Monburinon, P. Chertchom, T. Kaewkiriya, S. Rungpheung, S. Buya, and P. Boonpou, “Prediction of prices for used car by using regression models,” *Proceedings of 2018 5th International Conference on Business and Industrial Research: Smart Technology for Next Generation of Information, Engineering, Business and Social Science, ICBIR 2018*, pp. 115–119, Jun. 2018, doi: 10.1109/ICBIR.2018.8391177.

[20] P. Gajera, A. Gondaliya, and J. Kavathiya, “OLD CAR PRICE PREDICTION WITH MACHINE LEARNING,” *International Research Journal of Modernization in Engineering Technology and Science www.irjmets.com @International Research Journal of Modernization in Engineering*, pp. 2582–5208, Accessed: May 29, 2022. [Online]. Available: www.irjmets.com

[21] M. Listiani, R. Möller, M. Morlock, and S. Lessmann, “Support Vector Regression Analysis for Price Prediction in a Car Leasing Application,” 2009.

[22] T. Quang, N. Minh, D. Hy, and M. Bo, “Housing Price Prediction via Improved Machine Learning Techniques,” *Procedia Computer Science*, vol. 174, pp. 433–442, 2020, doi: 10.1016/J.PROCS.2020.06.111.

[23] Y. Ren and Y. Ren, “A Framework of Petroleum Information Retrieval System Based on Web Scraping with Python,” *2018 15th International Conference on Service Systems and Service Management, ICSSSM 2018*, Sep. 2018, doi: 10.1109/ICSSSM.2018.8465013.

[24] S. De and S. Sirisuriya, “A Comparative Study on Web Scraping,” 2015.

[25] S. Thivaharan, G. Srivatsun, and S. Sarathambekai, “A Survey on Python Libraries Used for Social Media Content Scraping,” *Proceedings - International Conference on Smart Electronics and Communication, ICOSEC 2020*, pp. 361–366, Sep. 2020, doi: 10.1109/ICOSEC49089.2020.9215357.

[26] D. Pratiba, M. S. Abhay, A. Dua, G. K. Shanbhag, N. Bhandari, and U. Singh, “Web Scraping and Data Acquisition Using Google Scholar,” *Proceedings 2018 3rd International Conference on Computational Systems and Information Technology for Sustainable Solutions, CSITSS 2018*, pp. 277–281, Dec. 2018, doi: 10.1109/CSITSS.2018.8768777.

[27] Fatmasari, Y. N. Kunang, and S. D. Purnamasari, “Web Scraping Techniques to Collect Weather Data in South Sumatera,” *Proceedings of 2018 International Conference on Electrical Engineering and Computer Science, ICECOS 2018*, pp. 385–390, Jan. 2019, doi: 10.1109/ICECOS.2018.8605202.

[28] M. Rakhra *et al.*, “Crop Price Prediction Using Random Forest and Decision Tree Regression:-A Review,” *Materials Today: Proceedings*, Apr. 2021, doi: 10.1016/J.MATPR.2021.03.261.

[29] A. Pandey, V. Rastogi, and S. Singh, “‘Car’s Selling Price Prediction using Random Forest Machine Learning Algorithm.’”, Accessed: May 31, 2022. [Online]. Available: https://ssrn.com/abstract=3702236

[30] S. Sushmita *et al.*, “Population cost prediction on public healthcare datasets,” *ACM International Conference Proceeding Series*, vol. 2015-May, pp. 87–94, May 2015, doi: 10.1145/2750511.2750521.