VERİ MADENCİLİĞİ PROJE İLERLEME RAPORU

(Project Outline Report)

Ders : FET445 - Veri Madenciliği

Konu: İkinci El Araç Fiyat Tahminlemesi (Car Price Prediction)

Bölüm: Yazılım Mühendisliği

Grup: AutoData Minds

Grup Üyeleri

İsim Soyisim	Numara	E-Mail
Ramazan Bozkurt	22040301027	ramazannbozkurrtt@outlook.com
Ömer Utku Aktemur	22040301043	aktemurutku@gmail.com
Erkan Yasin Yaman	22040301038	erkan.yasin00@gmail.com
Nurettin Kaplan	22040301031	nurettinkaplaan@gmail.com
Aslı Erbaşı	22040101040	asliierbasi@gmail.com

Repo Link: https://github.com/erbasiasli/VeriMadenciligiProje

1.6 multijet repo linki : https://github.com/RamazanBozkurrtt/1.6Multijet-VeriMadenciligiProje

2. Problem Tanımı

- **İş/Bilimsel Soru:** İkinci el araç piyasasında fiyatlar; marka, model, yıl, motor hacmi, yakıt tipi ve kilometre gibi birçok değişkene bağlı olarak şekillenmektedir. Bu projenin temel sorusu şudur: "Bir aracın teknik özellikleri ve kullanım geçmişi bilindiğinde, makine öğrenmesi algoritmaları aracın adil piyasa değerini ne kadar düşük bir hata payı ile tahmin edebilir ve araçlar teknik özelliklerine göre hangi doğal pazar segmentlerine (kümelere) ayrılır?
- **Görev Türü :**
  - **Regresyon(Prediction):** Araç fiyatının sayısal tahmini.
  - **Kümeleme(Clustering):** Araçların özelliklerine göre segmentlere ayrılması.
- **Hedef Değişken:** Price (Araç Fiyatı)
  - **Birim:** GBP (£)
  - **Etki Alanı:** Sürekli sayısal değişken (Pozitif tam sayı)
- **Başarı Kriterleri:**
  - Modelin açıklayıcılık katsayısı (R^2 Score) ≥ 0.85
  - Hata Kareler Ortalamasının Karekökü (MAE) değerinin minimize edilmesi.

3. Proje Yönetimi

Kilometre Taşları ve Zaman Çizelgesi:

- 1. Hafta (20-27 Ekim): Veri seti seçimi (Kaggle - UK Used Car Dataset) ve literatür taraması. (Tüm Grup)
- 2. Hafta (3-10 Kasım): Veri birleştirme, temizleme ve Keşifsel Veri Analizi (EDA).
- 3. Hafta: Eksik veri stratejisinin belirlenmesi ve uygulanması ve her üyenin kendi belirlediği 2 farklı Base Model ve 1 Kümeleme Algoritması ile model geliştirme süreçleri. (Bireysel Çalışmalar)
- 4. Hafta: Performans analizi, modellerin RMSE/R2 skorlarının karşılaştırılması ve en iyi modelin seçimi. Proje raporunun ve sunumun hazırlanması.

Roller ve Sorumluluklar:

- **Ramazan Bozkurt:**
  - Kümeleme: Agglomerative Clustering (Hiyerarşik Kümeleme)
  - Modeller: KNN Regressor ve Decision Tree Regressor
- **Erkan Yasin Yaman:**
  - Kümeleme: K-NN Clustering
  - Modeller: Linear Regression ve Lasso Regression(Threshold,Scaling)
- **Ömer Utku Aktemur:**

- Kümeleme: K-Means Clustering
- Modeller: Ridge Regression ve Support Vector Regression (SVR)

- **Aslı Erbaşı:**
  - Kümeleme: DBSCAN
  - Modeller: Decision Tree ve Bayesian Ridge
- **Nurettin Kaplan:**
  - Kümeleme: GMM (Gaussian Mixture Model)
  - Modeller: ElasticNet Regresyonu ve SGDRegressor

Çıktılar:

- Final Proje Raporu (PDF)
- Jupyter Notebook Kod Dosyaları
- Temizlenmiş Veri Seti
- Temizlenmiş ve Kümelenmiş Veri Seti

4. İlgili Çalışmalar (Mini Literatür İncelemesi)

- **Referans 1:** Pudaruth, S. (2014). "Predicting the Price of Used Cars using Machine Learning Techniques". Bu çalışmada yazar, KNN, Çoklu Doğrusal Regresyon ve Karar Ağaçlarını karşılaştırmış, Karar Ağaçlarının en iyi sonucu verdiğini belirtmiştir.
- **Referans 2:** Gegic, E. et al. (2019). "Car Price Prediction using Machine Learning". Yapay Sinir Ağları ve Random Forest kullanılarak %90 üzeri başarı elde edilmiştir.
- **Karşılaştırma ve Farkımız:** Mevcut literatürde genellikle eksik veriler ortalama ile doldurulup geçilmektedir. Bizim projemizin farkı, eksik kategorik verileri (Marka/Model) basit yöntemler yerine "Decision Tree Classifier" kullanarak diğer teknik özelliklerden tahmin edip doldurmamızdır (Smart Imputation). Ayrıca Feature Selection (SelectKBest) yönteminin, Boyut İndirgeme (PCA) yöntemine göre temel modeller üzerindeki etkisi spesifik olarak incelenmiştir.

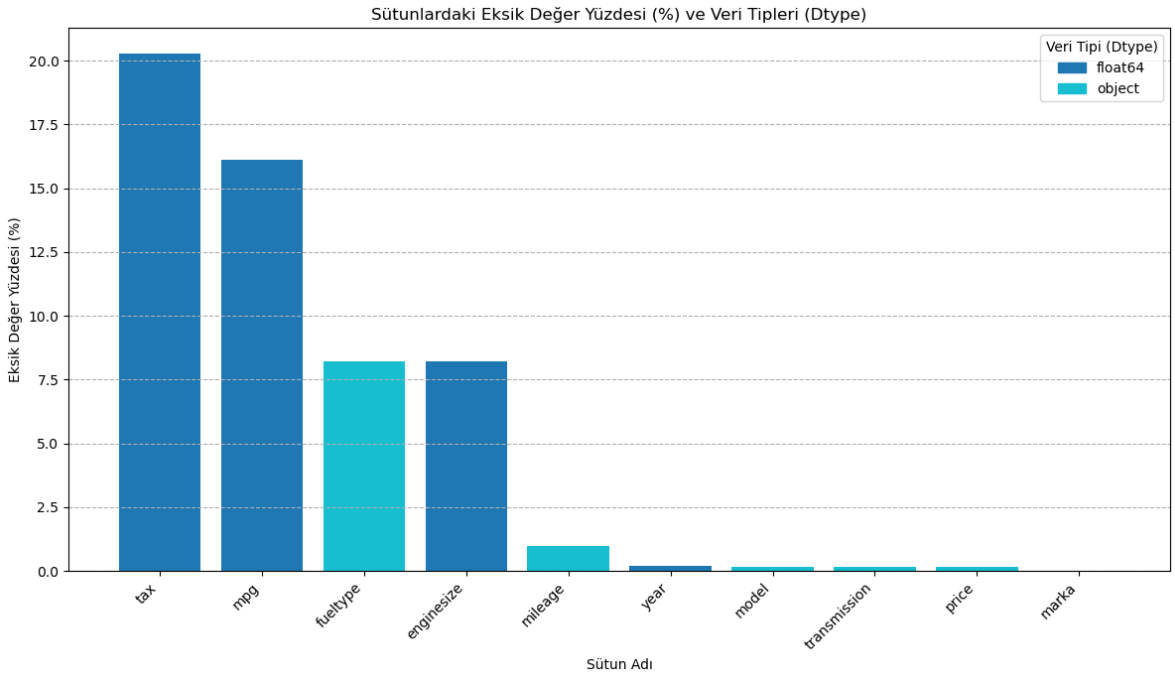
5. Veri Tanımı ve Yönetimi

- **Veri Seti:** "100,000 UK Used Car Data set" (Kaggle). Audi, BMW, Ford, VW gibi markaların verileri birleştirilmiştir.
- **Bağlantı:** https://www.kaggle.com/datasets/adityadesai13/used-car-dataset-ford-and-mercedes
- **Veri Şeması:**
  - model, transmission, fuelType: Kategorik (String)
  - year, mileage, tax, mpg, engineSize: Sayısal (Float/Int)
  - price: Hedef Değişken (Integer)

- **Boyut:** Orijinal veri setinde 100.000'den fazla satır bulunmaktadır. Ancak projenin hesaplama verimliliği açısından, tüm markaları kapsayan 30.000 satırlık rastgele ve tekrarlanabilir (random_state=42) bir örneklem oluşturulmuştur. 10 Sütun (9 Orijinal özellik + 1 Türetilmiş brand özelliği). (30.000x10)
- **Veri Erişim Planı:** Veriler CSV formatında GitHub deposunda saklanmaktadır.
- **Etik ve Gizlilik:** Veri seti halka açık (public domain) olup, kişisel tanımlayıcı bilgi (isim, plaka, şasi no) içermemektedir. Önyargı riski olarak, veri seti sadece İngiltere pazarını kapsadığı için modeller Türkiye pazarında doğrudan kullanılamayabilir.

6. Keşifsel Veri Analizi (Exploratory Data Analysis)

- **Veri Kalitesi:** tax, tax(£) ve mileage, mileage2 gibi tekrar eden ve eksik sütunlar tespit edilmiştir.
- **Dağılımlar:** Hedef değişken price sağa çarpık (right-skewed) bir dağılım göstermektedir (Az sayıda çok pahalı araç, çok sayıda orta segment araç).
- **İlişkiler:** car_age (Araç yaşı) ile price arasında güçlü negatif korelasyon, engineSize ile price arasında pozitif korelasyon gözlemlenmiştir.



7. Veri Hazırlama Planı

Bu bölümde, grup üyelerinin veri hazırlama aşamasında uyguladıkları ortak ve farklılaşan (bireysel) stratejiler detaylandırılmıştır.

7.1. Ortak Temizlik İşlemleri

- **Temizleme:** Farklı isimlerdeki benzer sütunlar (tax ve tax(£)) birleştirilip (fillna) tekilleştirilmiştir.
- **Özellik Mühendisliği:** year değişkeninden car_age (Araç Yaşı) türetilmiştir.

7.2. Bireysel Ayrışma Noktaları (Differentiation Points)

A. Ramazan Bozkurt'un Yaklaşımı:

- **Veri Temizleme ve Kalite Kontrolü (Data Cleaning):**
    - Regex ile Veri Tipi Düzeltme: Price ve Mileage sütunlarındaki para birimi sembolleri (£,(,,)) Regex ile temizlenerek veriler sayısal formata dönüştürülmüştür.
    - Sütun Standardizasyonu: Yazım tutarsızlıklarını gidermek için tüm sütun adları standartlaştırılmış ve özel karakterlerden arındırılmıştır.
    - Gereksiz Sütunların Atılması: Tekrarlayan veya analize gürültü katan sütunlar (reference, tax_gbp vb.) veri setinden çıkarılmıştır.
    - Hedef Değişken Temizliği: Hedef değişken olan price sütununda eksik veya hatalı veri bulunan satırlar, eğitim sürecini bozmaması adına veri setinden silinmiştir.

- **İmputasyon Stratejisi (Eksik Veri Tamamlama):**
    - Birleştirme (Coalesce) Mantığı: Farklı sütunlara dağılmış benzer veriler (mileage varyasyonları) birleştirilerek veri kaybı önlenmiştir.
    - Model Tabanlı İmputasyon (Kategorik): Basit "Mod" yöntemi yerine, veri setindeki desenleri öğrenen bir Decision Tree Classifier kullanılmıştır.
    - İstatistiksel İmputasyon (Sayısal): Sayısal eksik veriler, aykırı değerlerden etkilenmemesi amacıyla Medyan (Ortanca) stratejisi ile doldurulmuştur.

- **Veri Dönüşümü (Transformation) ve Kodlama:.**
    - One-Hot Encoding: KNN ve Decision Tree modellerinin kategorik veriyi işleyebilmesi için Marka, Model gibi değişkenler One-Hot Encoding ile sayısal matrislere dönüştürülmüştür.
    - Ölçekleme (Scaling) Deneyi (A/B Testi): Mesafe temelli KNN için zorunlu olan StandardScaler'ın etkisi (Ham Veri vs. Ölçekli Veri) deneysel olarak test edilmiş ve Karar Ağaçlarının ölçekten bağımsız çalıştığı kanıtlanmıştır.

- **Özellik Mühendisliği (Feature Engineering) ve Zenginleştirme:**
    - Hiyerarşik Kümeleme ile Özellik Türetimi (Agglomerative Clustering): Araçların gizli pazar segmentlerini (Lüks, Ekonomik) keşfetmek için Agglomerative Clustering uygulanmış; elde edilen küme etiketleri (segment_cluster), regresyon modellerine yeni bir öznitelik olarak eklenmiştir.
    - Amaç: Modelin, "Lüks", "Ekonomik", "Spor" gibi gizli pazar segmentlerini öğrenmesini kolaylaştırmak.
    - Uygulama: Küme etiketleri (segment_cluster), regresyon modellerinin fiyat tahminini iyileştirmesi için girdi değişkeni olarak kullanılmıştır.

- Alan Bilgisi (Domain Knowledge): Araçların üretim yılı (year) yerine, ikinci el piyasasındaki değer kaybını (depreciation) daha iyi ifade eden Araç Yaşı (car_age) özelliği türetilmiştir.

- **Özellik Seçimi (Feature Selection):**
  - Filter Method: Modelin yorumlanabilirliğini korumak adına Filter Method (SelectKBest, k=15) uygulanmıştır.
  - Gözlem: Bu işlemin, veri setinin yüksek bilgi yoğunluğu nedeniyle bazı senaryolarda Bilgi Kaybı yaratarak model performansını düşürdüğü gözlemlenmiştir.

B. Erkan Yasin Yaman'ın Yaklaşımı:

- Variance Threshold (Varyans Eşiği): PCA gibi veriyi dönüştürmek yerine, ayırt ediciliği düşük olan (varyansı 0.01'in altındaki) özellikleri eleyerek modelin karmaşıklığı azaltılmıştır.
- k-NN ile Özellik Mühendisliği (Clustering-based Feature Engineering): Projenin en yenilikçi adımıdır. Gözetimsiz öğrenme mantığıyla çalışan k-Nearest Neighbors (k-NN) algoritması kullanılarak araçlar fiziksel özelliklerine göre *Ekonomik*, *Orta* ve *Lüks* olmak üzere 3 segmente ayrılmıştır. Bu "Tahmini Segment" bilgisi, regresyon modellerine yeni bir öznitelik (feature) olarak eklenmiştir.
- Feature Scaling (Ölçekleme): Özellikle Lasso Regresyon ve k-NN mesafe tabanlı çalıştığı için, veri StandardScaler ile ölçeklenmiştir.

İki farklı regresyon algoritması (Linear & Lasso), 4 farklı senaryoda test edilmiştir:

- **A. Linear Regression: Temel regresyon modelidir.**
  - *Senaryo:* Saf Hali, Sadece Threshold, Sadece Kümeleme ve Full Paket (Threshold + Kümeleme).
  - *Sonuç:* Saf model %77 başarı gösterirken, k-NN segmentasyonu eklendiğinde R^2 skoru %78.16'ya yükselmiştir.
- **B. Lasso Regression (L1 Regularization): Özellik seçimi yapabilen, gereksiz katsayıları sıfırlayan modeldir.**
  - *Senaryo:* Saf Hali (Başarısız), Scaling + Threshold (Standart) ve Hibrit Model (Scaling + Threshold + k-NN Segmentasyon).
  - *Sonuç:* Ölçeklenmemiş veride başarısız olan Lasso, hibrit yapıda kullanıldığında Linear Regression ile yarışır bir performans sergileyerek yüksek başarı oranlarına ulaşmıştır.

C. Ömer Utku Aktemur'un Yaklaşımı:

1. Veri Dönüşümü ve Ölçekleme:

- Kullandığım uzaklık ve varyans temelli modellerin (Ridge, SVR, K-Means, PCA) hassasiyeti nedeniyle veri setine StandardScaler uygulanması zorunlu tutulmuştur. Kategorik değişkenler, sınıf sayılarına göre Label Encoding (model sütunu) ve One-Hot Encoding ile sayısal hale getirilmiştir.

2. Boyut İndirgeme (PCA):

- Özellik seçimi yerine, değişkenler arasındaki çoklu bağlantı (multicollinearity) sorununu çözmek için PCA tercih edilmiştir. Veri setindeki varyansın %95'ini koruyacak şekilde boyut indirgeme uygulanmıştır.

3. Model Seçimi:

- Ridge Regresyon: Doğrusal ilişkileri modellemek ve L2 regularizasyonu ile aşırı öğrenmeyi engellemek için.
- SVR: Doğrusal olmayan (non-linear) karmaşık yapıları RBF Kernel ile yakalamak için.
- K-Means: Fiyat tahmininden bağımsız pazar segmentasyonu yapmak ve bu bilgiyi tahmin modellerine ek özellik olarak sunmak için.

4. Deneysel Tasarım (4 Senaryo):

- Farklı tekniklerin performans etkisini ölçmek için modeller 4 aşamalı test edilmiştir:
- Base Model: Standartlaştırılmış ham veri (Referans).
- Sadece PCA: Gürültüsü azaltılmış ve boyutu indirgenmiş veri.
- Kümeleme Destekli: Ham veriye "Pazar Segmenti" bilgisinin eklendiği veri.
- Hibrit (PCA + Kümeleme): Hem boyutun indirgendiği hem de segmentasyon bilgisinin eklendiği en gelişmiş yapı.

D. Aslı Erbaşı'ın Yaklaşımı:

Veri Temizliği ve Format Standardizasyonu:

Analiz öncesinde veri setindeki en büyük engel, *price* ve *mileage* sütunlarındaki standart dışı karakterlerdi. Pandas ile okuma yaptıktan sonra, Regex fonksiyonları kullanarak bu alanlardaki para birimi işaretlerini (£) ve virgülleri temizleyip sayısal formata dönüştürdük. Ayrıca veri bütünlüğünü sağlamak adına, analiz sonuçlarını saptırabilecek 301 adet mükerrer (duplicate) kaydı tespit edip veri setinden çıkardık.

Aykırı Değer (Outlier) Yönetimi:

Modelin (özellikle regresyon algoritmalarının) uç değerlerden negatif etkilenmemesi için mantıksal sınırlar belirledik. 100£ altındaki hurda niteliğindeki araçlar ile 200.000£ üzerindeki ekstrem lüks araçları ve 1995 öncesi modelleri eğitim setine dahil etmedik. Bu sayede modelin genel piyasa trendlerini öğrenmesini kolaylaştırdık.

Eksik Veri Stratejisi (Imputation):

Veri kaybını önlemek amacıyla satır silmek yerine SimpleImputer yöntemini tercih ettik. Eksik verileri, ilgili sütunun ortalamasıyla (mean) doldurarak veri dağılımını bozmadan setin tamamını modellenebilir hale getirdik.

Öznitelik Mühendisliği (Feature Engineering):

- Araç Yaşı: "Year" bilgisi yerine, 2025 yılını baz alarak aracın yaşını hesapladık; bu, ikinci eldeki değer kaybını modellemek için daha etkili bir metrik oldu.

- Kullanım Yoğunluğu: Kilometreyi yaşa bölerek *Avg_Km_Per_Year* (Yıllık Ortalama Kullanım) özelliğini türettik. Böylece model, "yeni ama çok kullanılmış" araç ile "eski ama az kullanılmış" aracı ayırt edebilir hale geldi.

Kodlama ve Özellik Seçimi (Encoding & Selection):

Kategorik verileri (Marka, Vites vb.) Decision Tree ve Bayesian Ridge modellerinin işleyebileceği sayısal matrislere dönüştürmek için One-Hot Encoding uyguladık. Ancak bu işlem sütun sayısını 200'ün üzerine çıkardığı için, işlem maliyetini düşürmek adına *SelectKBest* algoritmasını kullandık. Fiyat üzerinde istatistiksel olarak en etkili 10 özelliği seçerek modelleri sadeleştirdik.

6. Algoritma Seçimi ve Modelleme Stratejisi:

- Regresyon (Fiyat Tahmini): Fiyat tahmini için temel (base) model olarak Bayesian Ridge ve Decision Tree algoritmalarını karşılaştırdık. Decision Tree modelinin ham haliyle bile daha yüksek başarı (0.89 R2) göstermesi üzerine, bu modeli GridSearchCV ile optimize ederek (Overfitting'i önlemek için derinlik ayarı vb.) final model olarak belirledik.
- Kümeleme (Segmentasyon): Araçları özelliklerine göre gruplamak için K-Means gibi merkeze zorlayan yöntemler yerine, yoğunluk tabanlı çalışan DBSCAN algoritmasını tercih ettik. Bu sayede veri setindeki doğal yoğunluk kümelerini (segmentleri) yakaladık ve gürültü (noise) verilerini ayıkladık.

E.Nurettin Kaplan'ın Yaklaşımı:

Veri Hazırlığı ve Ön İşleme

Kodlama aşamasına veriyi temizleyerek başladım. İlk baktığımda veri setinde tutarsızlıklar ve tekrar eden kayıtlar vardı, bunları ayıkladım. Fiyat ve kilometre gibi sayısal olması gereken sütunlardaki para birimi simgelerini temizleyip sayısal formata çevirdim.

- Eksik Veriler: Eksik alanları doldururken veri dağılımını bozmamaya dikkat ettim. Sayısal verileri medyan, kategorik olanları ise mod (en sık tekrar eden) değer ile doldurdum.
- Öznitelik Üretimi: Modelin işini kolaylaştırmak için mevcut veriden yeni bilgiler çıkardım. Üretim yılından Araç Yaşı'nı hesapladım ve kilometreyi yaşa bölerek Yıllık Ortalama Kilometre bilgisini türettim. Bu yeni özellikler tahmin gücünü artırdı.
- Dönüştürme (Encoding): Kategorik verileri (Marka, Model vb.) makinenin anlayacağı dile çevirmek için One-Hot Encoding yaptım. Bu işlem sütun sayısını artırsa da modelin markalar arasındaki farkı daha iyi anlamasını sağladı.

Modelleme Yaklaşımı

- ElasticNet Regresyonu: Bu modeli seçmemin sebebi, Lasso ve Ridge'in en iyi yönlerini birleştirmesiydi. Verimizde çok fazla sütun (özellik) olduğu için aşırı öğrenmeyi (overfitting) engellemek adına bu hibrit yapıyı kullandım.
- Huber Regressor: İkinci el araç piyasasında çok uçuk fiyatlar (outlier) olabiliyor. Huber modeli, karesel hata yerine mutlak hatayı kullanarak bu aykırı değerlerin modelin genelini bozmasını engellediği için tercih ettim.

Optimizasyon: GridSearchCV ile en iyi parametrelerini (alpha, epsilon vb.) arattım. Ayrıca modelleri 3 farklı senaryoda yarıştırdım:

1. Ham Veri: Tüm özelliklerin kullanıldığı durum.
2. Öznitelik Seçimi (RFE): İstatistiksel olarak en değerli 50 özelliği seçerek denedim.
3. Boyut İndirgeme (SVD): Veriyi sıkıştırıp boyutu azalttım.

Pazar Segmentasyonu (Kümeleme)

Araçları fiyatlarına ve özelliklerine göre gruplamak için Gaussian Mixture Models (GMM) algoritmasını kullandım. K-Means genelde yuvarlak kümeler ararken, GMM verinin dağılımına olasılıksal baktığı için daha esnek ve doğru bir segmentasyon sağladı. Analiz sonucunda araçları "Ekonomik", "Orta Sınıf", "Lüks" gibi 4 farklı segmente ayırdık.

8. Modelleme Planı

Bu bölümde, geliştirilen temel modellerin performans sonuçları, geliştiricilerine göre ayrılmıştır.

8.1. Baseline (Referans) Model Stratejisi

Modellerin başarısını ölçümleyebilmek için öncelikle bir "Dumb Baseline" stratejisi belirlenmiştir.

- Strateji: Veri setindeki ortalama fiyatı (Mean) tahmin eden basit bir yaklaşım referans noktası olarak kabul edilecektir. Geliştirdiğimiz modellerin bu referans noktasından ve birbirlerinden ne kadar iyi performans gösterdiği analiz edilecektir.

8.2. Aday Modeller ve Seçim Gerekçeleri

Grup üyeleri, veri setindeki farklı yapıları (doğrusal ve doğrusal olmayan ilişkiler) test etmek amacıyla farklı algoritmalar seçmiştir

A. Ramazan Bozkurt - KNN ve Decision Tree Analizi

Baseline Model:

- Projede "Scale Yok (Ham Veri)" senaryosu, yinelemeli (iterative) model geliştirme sürecinin başlangıç noktası ve Baseline (Taban) Model olarak kurgulandı. Parametre optimizasyonu yapılmadan kurulan bu saf modeller, ileri ön işleme tekniklerinin (Scaling, Kümeleme, Feature Selection) modele katkısını ölçmek için referans noktası oldu.

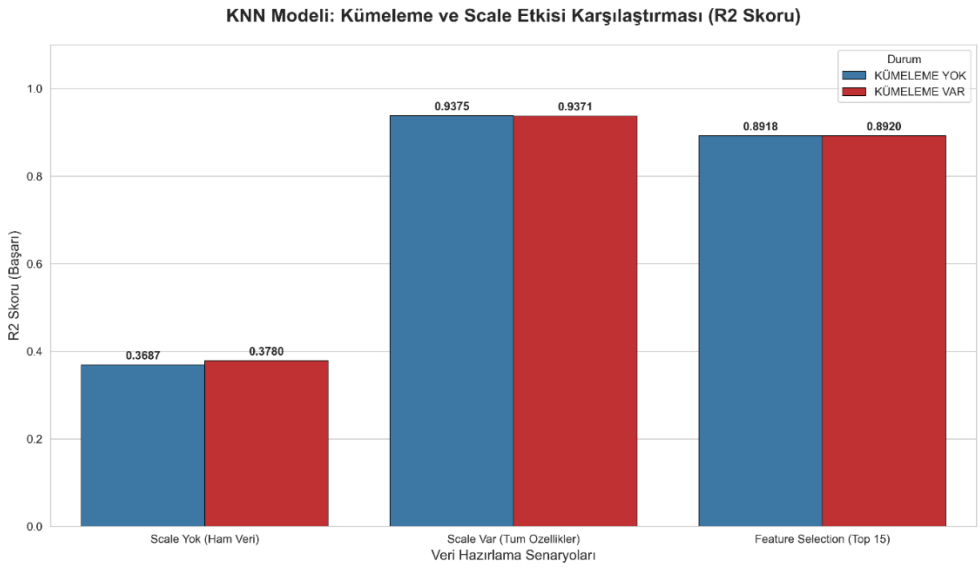
Aday Modeller :

1. **K-Nearest Neighbors (KNN) Regressor:**
   - Neden: Araç piyasasında fiyat belirleme mantığı genellikle "benzer araçların fiyatına bakma" ilkesine dayanır. KNN, mesafe tabanlı yapısıyla bu "benzerlik" ilkesini en iyi simüle eden algoritmadır.

- **Performans Tablosu:**

Model	KÜMELEME	Veri Durumu	MAE (Ort. Hata)	R2 Score (Başarı)
KNN	YOK	Scale Yok (Ham Veri)	5488.283394	0.368714

KNN	YOK	Scale Var (Tum Ozellikler)	1412.279433	0.937450
KNN	YOK	Feature Selection (Top 15)	1853.360901	0.891799
KNN	VAR	Scale Yok (Ham Veri)	5441.806468	0.378018
KNN	VAR	Scale Var (Tum Ozellikler)	1420.301926	0.937124
KNN	VAR	Feature Selection (Top 15)	1845.232304	0.891975



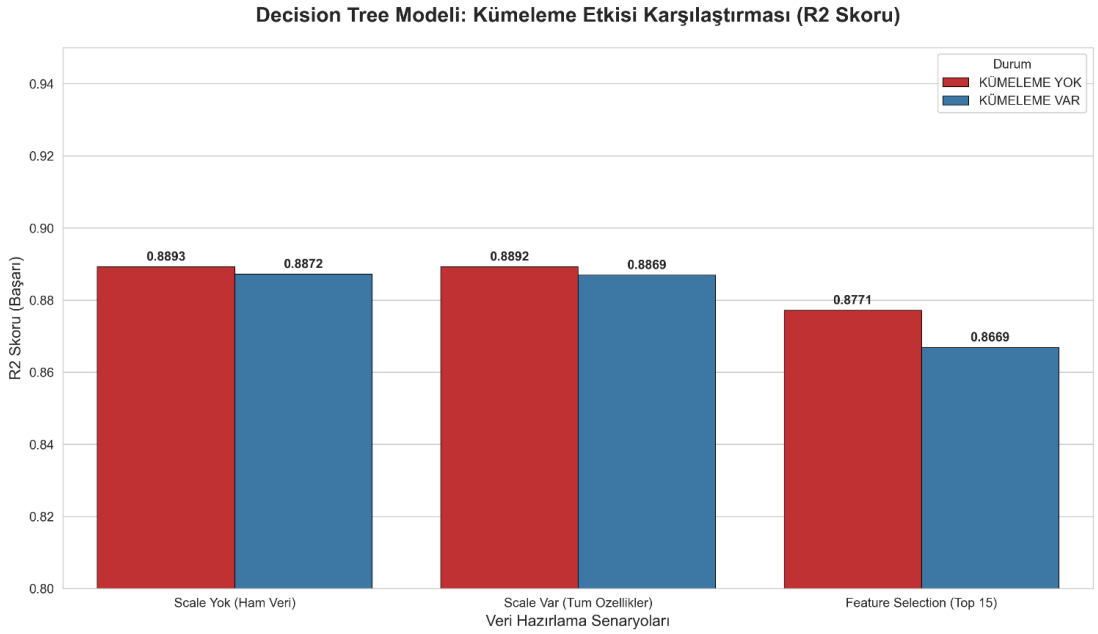
2. Decision Tree (Karar Ağacı) Regressor:

- *Neden:* Verisetindeki doğrusal olmayan (non-linear) ilişkileri yakalayabilmesi ve if-then kuralları ile insan tarafından yorumlanabilir (interpretable) sonuçlar üretmesi nedeniyle tercih edilmiştir.

- Yapısal Tercih : Bu çalışmada kullanılan Karar Ağacı, varsayılan (default) ayarların aksine, aşırı öğrenmeyi (overfitting) engellemek ve modelin genelleme yeteneğini artırmak amacıyla sınırlandırılmıştır (Pruning). Ağacın sonsuz derinleşmesine izin verilmemiş, derinlik (max_depth) ve yaprak başına düşen örnek sayısı (min_samples_leaf) optimize edilerek kontrol altında tutulmuştur.

- **Performans Tablosu:**

Model	KÜMELEME	Veri Durumu	MAE (Ort. Hata)	R2 Score (Başarı)
Decision Tree	YOK	Scale Yok (Ham Veri)	2072.299491	0.889252
Decision Tree	YOK	Scale Var (Tum Ozellikler)	2073.177752	0.889192
Decision Tree	YOK	Feature Selection (Top 15)	2160.075671	0.877114
Decision Tree	VAR	Scale Yok (Ham Veri)	2114.905589	0.887193
Decision Tree	VAR	Scale Var (Tum Ozellikler)	2117.165627	0.886859
Decision Tree	VAR	Feature Selection (Top 15)	2224.584477	0.866883



- **Değerlendirme ve Analiz:**
  - Genel Başarı ve Model Seçimi: Yapılan kapsamlı deneylerde KNN (Scale Var - Tüm Özellikler) senaryosu 0.937 R2 skoru ile en yüksek başarıyı göstermiştir. KNN algoritmasının yerel benzerlikleri yakalamadaki başarısı, araç fiyatlandırma probleminin doğasıyla tam uyum sağlamıştır.
  - Ölçeklemenin Kritik Etkisi:
    - KNN İçin: Ölçekleme yapılmadığında 0.36 gibi çok düşük bir R2 skoru veren KNN, StandardScaler uygulandığında başarıyı 0.93 seviyesine yükseltmiştir. Bu durum, mesafe temelli modellerin kanıtıdır.
    - Decision Tree İçin: Karar Ağaçlarının verinin ölçeğinden bağımsız (scale-invariant) çalıştığı doğrulanmıştır (Scale Var/Yok R2 skoru ~0.88).
  - Kümeleme (Clustering) Etkisi ve Redundancy: Agglomerative Clustering ile üretilen Cluster ID bilgisinin eklenmesi, performansta belirgin bir artış sağlamamıştır. Bunun nedeni, küme bilgisinin temel özellikler (Yıl, Motor, KM) tarafından zaten açıklanabilmesi ve bu bilginin model için yüksek korelasyonlu (redundant) kalmasıdır. Bu, DT'nin ham veriyi ezberlemeden en iyi kuralı bulduğunu göstermiştir.
  - Decision Tree Regression Üzerinde Scale ve Kümelemenin negatif etkili olması nedeni
    - Sonuç: Bu deney, Decision Tree modelinin ham veri (Raw Data) üzerindeki karmaşık ilişkileri yakalamada zaten oldukça yetenekli olduğunu ve bu veri seti özelinde ekstra özellik mühendisliğine (Feature Engineering) ihtiyaç

duymadığını kanıtlamıştır. En saf ve yalın model (Ham Veri), en iyi sonucu vermiştir.

- Hiper-Parametre Ayarlama:

  Modellerin temel performansını ölçmek adına aşağıdaki parametre konfigürasyonları kullanılmıştır:

  - KNN: Komşu sayısı n_neighbors=5 olarak belirlenmiştir.
  - Decision Tree (Özelleştirilmiş - pruning):
    - Modelin ezberlemesini önlemek için max_depth=10 (Maksimum derinlik) olarak sınırlandırılmıştır.
    - Gürültülü veriye aşırı duyarlılığı azaltmak için min_samples_leaf=4 (Bir yaprakta olması gereken minimum örnek) olarak ayarlanmıştır.
    - Bölünme kriteri için min_samples_split=10 kullanılmıştır.

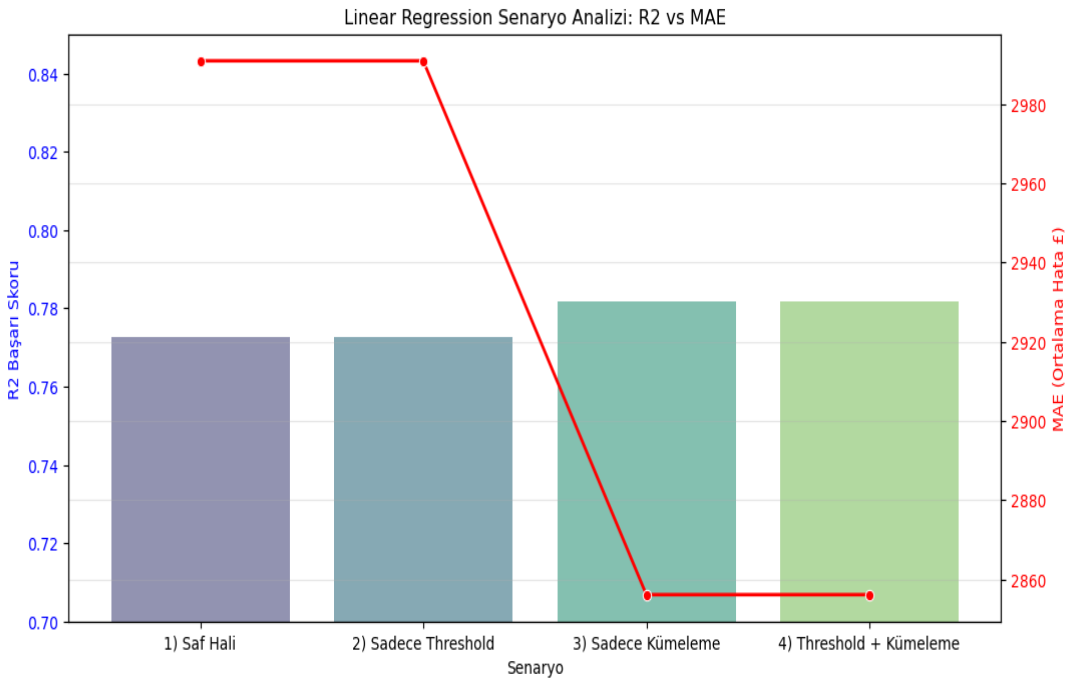
  - Feature Selection: Seçilecek özellik sayısı k=15 olarak sabitlenmiştir.

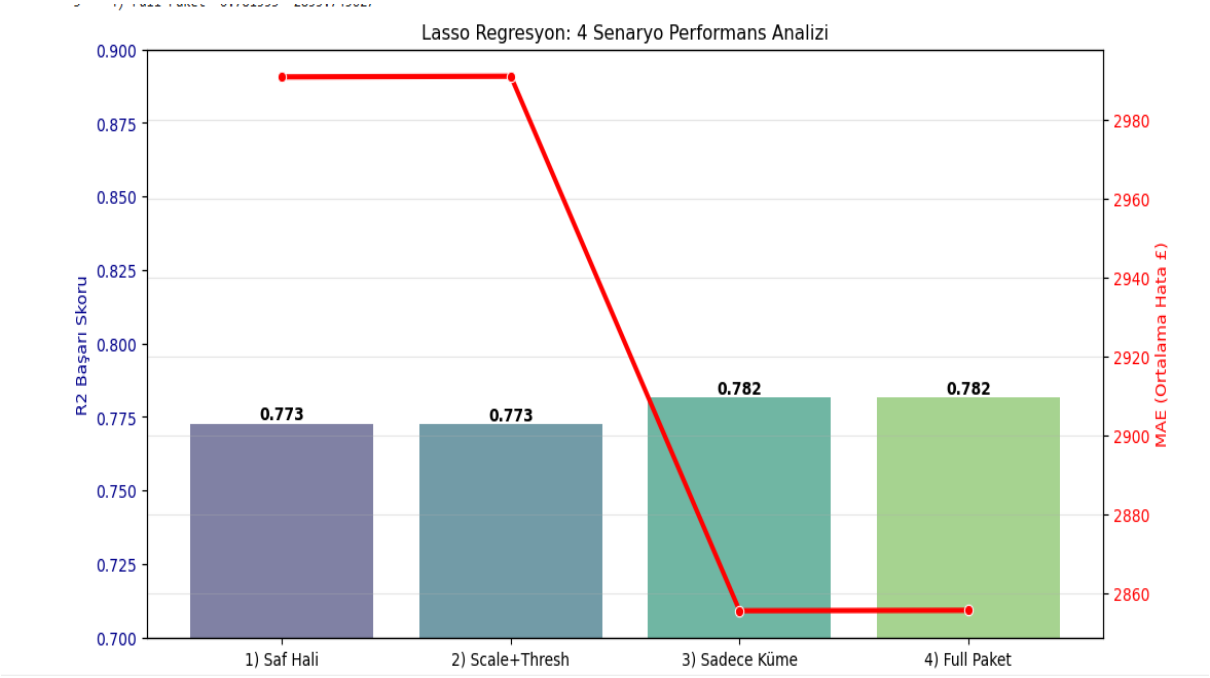
- Sınıf Dengesizliği Stratejisi (Class Imbalance):

  - Sınıf Dengesizliği Stratejisi: Proje Regresyon problemi olduğundan, Sınıf Dengesizliği kavramı teknik olarak geçerli değildir. Ancak, fiyat dağılımındaki aşırı uç değerler, Akıllı Eksik Veri Doldurma ve Standardizasyon teknikleri kullanılarak kontrol altında tutulmuştur.

B. Erkan Yaman - Linear Regression ve LassoRegresyon Analizi

- **Geliştirilen Modeller: Linear Regression ve Lasso Regression(Threshold,Scaling)**
- **Kullanılan Kümeleme: k-NN Clustering**





Yukarıdaki grafikler, veri ön işleme tekniklerinin model başarısı üzerindeki etkisini göstermektedir. İlk resim Linear Regression analizinde, k-NN algoritması ile üretilen "Segment" bilgisinin modele eklenmesiyle R^2 skorunda artış sağlandığı görülmektedir. İkinci Resim Lasso Regression analizinde ise, algoritmanın ölçekleme (scaling) işlemine olanbağımlılığı ve hibrit (Threshold + Kümeleme) yaklaşımın hata oranını (MAE) nasıl düşürdüğü görselleştirilmiştir. Her iki modelde de en iyi sonuçlar, istatistiksel eleme ve algoritmik özellik üretiminin bir arada kullanıldığı 4. Senaryo'da elde edilmiştir.

C. Ömer Utku Tarafından Geliştirilen Modeller:

Veri setindeki hem doğrusal hem de karmaşık ilişkileri kapsamak amacıyla aşağıdaki üç model seçilmiştir:

1. Ridge Regresyon (Doğrusal Model):

- Seçim Nedeni: Değişkenler arasındaki çoklu bağlantı (multicollinearity) sorununu aşmak ve L2 regularizasyonu ile aşırı öğrenmeyi (overfitting) engellemek için tercih edilmiştir.
- Beklenti: Fiyat üzerindeki doğrusal etkileri hızlı ve kararlı bir şekilde modellemesi hedeflenmiştir.

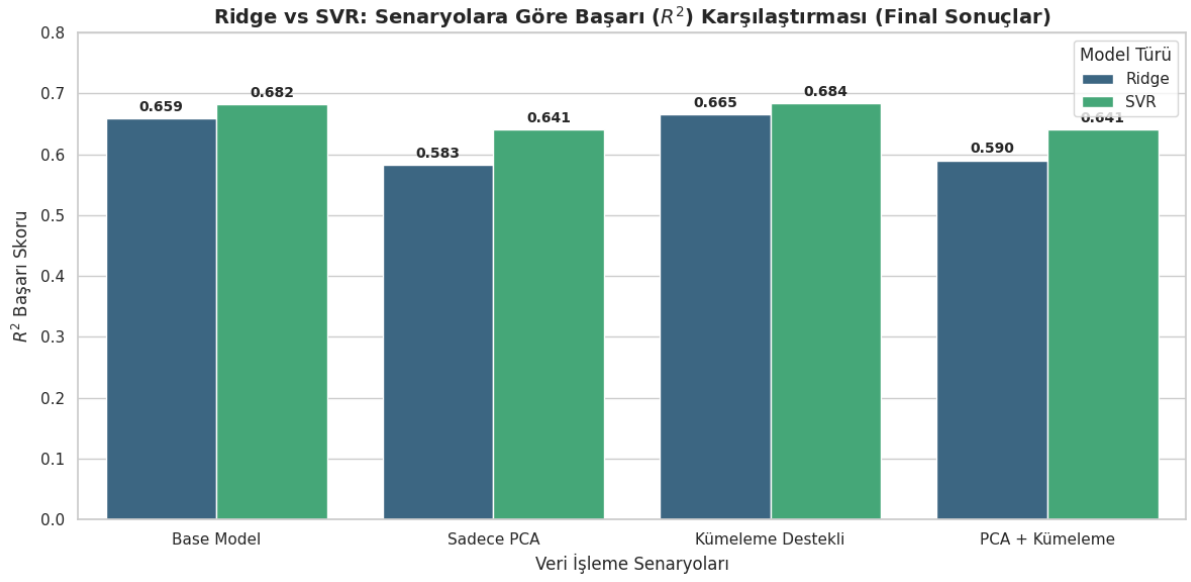
2. SVR - Destek Vektör Regresyonu (Kernel Tabanlı Model):

- Seçim Nedeni: Ridge modelinin yakalayamadığı doğrusal olmayan (non-linear) karmaşık fiyat desenlerini, RBF çekirdeği kullanarak modellemek için seçilmiştir.
- Beklenti: Büyük veri setlerinde eğitim maliyeti yüksek olsa da, PCA desteği ile karmaşık yapılarda en yüksek başarıyı göstermesi beklenmektedir.

3. K-Means (Kümeleme):

- Seçim Nedeni: Fiyat tahmininden bağımsız olarak, araçları teknik özelliklerine göre pazar segmentlerine (Ekonomik, Orta, Lüks) ayırmak için kullanılmıştır.
- Beklenti: Verinin gizli yapısını ortaya çıkarmak ve elde edilen "Segment Bilgisini" tahmin modellerine ek bir öznitelik olarak sunmaktır

Model	KMeans Kümeleme	Veri Durumu	MAE(Ort. Hata)	R^2 Score (Başarı)
Ridge	**YOK**	**PCA YOK (Ham Veri)**	**3412.05**	**0.6591**
Ridge	**YOK**	**PCA VAR**	**3905.62**	**0.5834**
Ridge	**VAR**	**PCA YOK (Ham Veri)**	**3332.46**	**0.6650**
Ridge	**VAR**	**PCA VAR**	**3844.57**	**0.5898**
SVR	**YOK**	**PCA YOK (Ham Veri)**	**2824.16**	**0.6823**
SVR	**YOK**	**PCA VAR**	**3123.19**	**0.6406**
SVR	**VAR**	**PCA YOK (Ham Veri)**	**2818.49**	**0.6840**
SVR	**VAR**	**PCA VAR**	**3123.34**	**0.6412**



Ridge ve SVR modellerinin 4 farklı senaryodaki performans kıyaslaması şu şekildedir:

1. Model Karşılaştırması (Ridge vs SVR):

- Bulgu: Tüm senaryolarda SVR, Ridge modeline göre üstünlük sağlamıştır (Base: 0.682 vs 0.659).
- Neden: Ridge sadece doğrusal ilişkileri modelleyebilirken; SVR, RBF çekirdeği sayesinde araç fiyatlarındaki marka-motor-yıl gibi karmaşık ve doğrusal olmayan etkileşimleri başarıyla yakalamıştır.

2. Boyut İndirgeme (PCA) Etkisi:

- Bulgu: PCA uygulandığında %5'lik varyans kaybına bağlı olarak performans düşüşü (SVR: -0.041) gözlemlenmiştir.
- Değerlendirme: Bu düşüş, eğitim süresinin kısalması ve model karmaşıklığının azalması karşılığında kabul edilebilir bir performans/hız takası (trade-off) olarak değerlendirilmiştir.

3. Kümeleme (Segmentasyon) Etkisi:

- Bulgu: Veriye "Pazar Segmenti" bilgisi eklendiğinde her iki model de en yüksek başarı skorlarına ulaşmıştır (SVR: 0.684).
- Yorum: Bu sonuç, araçların teknik özelliklerinin yanı sıra hangi pazar grubuna (Ekonomik, Lüks vb.) ait olduğu bilgisinin fiyat tahmininde kritik bir öznitelik olduğunu kanıtlamıştır.

Sonuç:

Deneyler sonucunda en uygun modelin SVR, en yüksek performans sağlayan veri yapısının ise Kümeleme Destekli senaryo (R^2=0.684$) olduğu tespit edilmiştir. PCA ise yüksek performans yerine verimlilik (hız) gerektiren durumlar için etkili bir alternatiftir.

D. Aslı Erbaşı - Decision Tree ve Bayesian Ridge Analizi

Veri setindeki hem doğrusal fiyat ilişkilerini hem de marka-model bazlı kırılımları daha iyi yakalayabilmek adına projemde şu üç temel yaklaşımı kullandım:

1. Bayesian Ridge Regresyon (İstatistiksel Doğrusal Model):

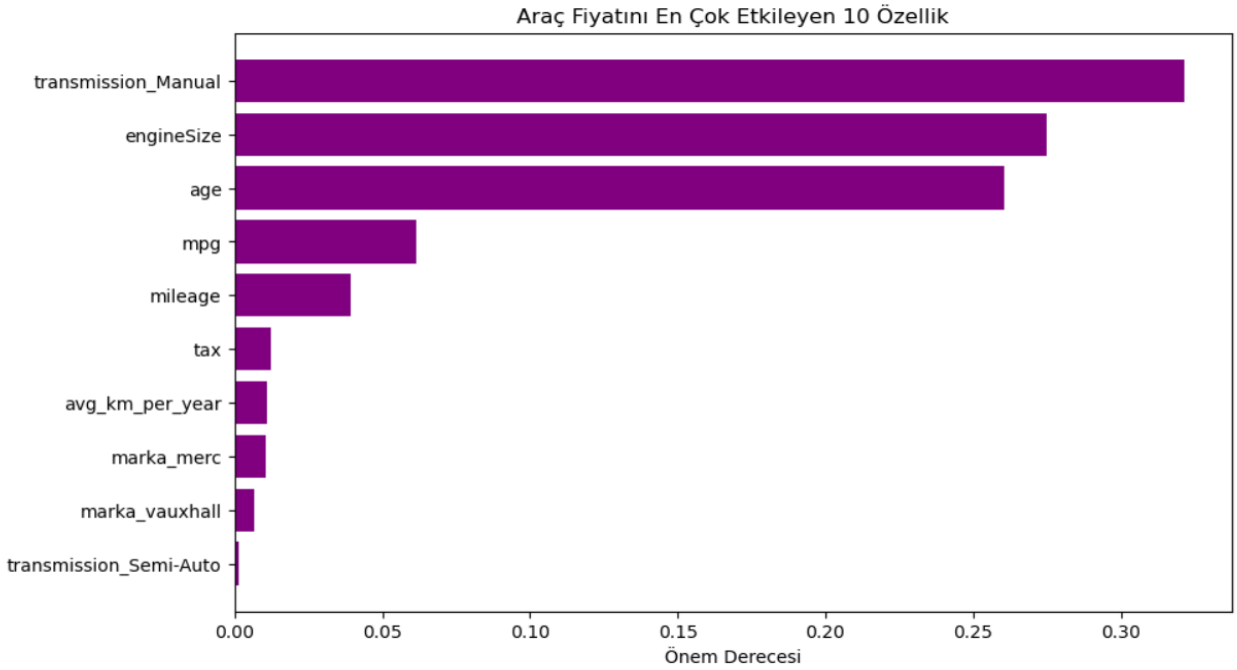
- Seçim Nedenim: Klasik lineer regresyonun aksine, bu model hem verideki belirsizliği (olasılıksal yaklaşım) hesaba kattığı hem de çok sayıda değişken olduğunda (One-Hot Encoding sonrası) katsayıları baskılayarak aşırı öğrenmeyi (overfitting) engellemeye yatkın olduğu için tercih ettim.
- Beklentim: Fiyat ile kilometre veya yaş gibi sayısal veriler arasındaki doğrusal ilişkiyi dengeli bir şekilde kurmasını bekliyordum.

2. Decision Tree Regresyon (Karar Ağacı Modeli):

- Seçim Nedenim: Araç fiyatları her zaman matematiksel bir düzlemde artmaz; bazen "markası Audi ise ve yılı 2018'den büyükse fiyat şudur" gibi keskin kurallar gerekir. Bayesian Ridge gibi doğrusal modellerin kaçırdığı bu karmaşık ve kural tabanlı desenleri yakalamak için bu modeli seçtim.
- Beklentim: Veri setimizdeki kategorik değişkenlerin (marka, vites tipi vb.) bolluğu nedeniyle, ağaç yapısının bu ayrımları daha iyi yaparak en yüksek başarıyı vereceğini öngördüm.

- **3. DBSCAN (Yoğunluk Tabanlı Kümeleme):**

- Seçim Nedenim: K-Means gibi klasik yöntemler yerine, araçları özelliklerine göre doğal yoğunluklarına ayırmak ve en önemlisi "aykırı" (outlier) yani pazarda istisna olan araçları tespit edebilmek için DBSCAN algoritmasını kullandım.
- Beklentim: Verideki gürültüyü ayıklamak ve araçları benzerliklerine göre (Ekonomik, Lüks vb.) segmentlere ayırmaktı.



Modellerin Performans Karşılaştırması ve Bulgularım:

Proje boyunca yaptığım denemelerde, iki ana regresyon modelini (Bayesian Ridge ve Decision Tree) farklı senaryolarda (Tüm özellikler, PCA, SelectKBest) yarıştırdım. Sonuçlar şu şekildedir:

1. Model Karşılaştırması (Bayesian Ridge vs. Decision Tree):

- Bulgu: Tüm senaryolarda Decision Tree (Karar Ağacı) modeli, Bayesian Ridge modeline göre bariz bir üstünlük sağladı. (Base Skorlar -> Decision Tree: 0.897 vs. Bayesian Ridge: 0.867).
- Neden: Bayesian Ridge, veriye düz bir çizgi veya düzlem oturtmaya çalışırken; Decision Tree, veriyi dallara ayırarak "Lüks araçlar" veya "Düşük kilometreli araçlar" gibi alt grupları çok daha iyi modelledi. Araç fiyat tahmininde özelliklerin birbirini kestiği noktalar (non-linear ilişkiler) çok fazla olduğu için ağaç tabanlı model daha başarılı oldu.

2. Boyut İndirgeme (PCA) ve Özellik Seçimi (SelectKBest) Etkisi:

- Bulgu: One-Hot Encoding sonrası elimde 202 özellik vardı. PCA ile bunu 173 bileşene indirdiğimde Decision Tree modelinin başarısında ufak bir düşüş (0.897 -> 0.876) gözlemledim. Ancak SelectKBest ile sadece en iyi 10 özelliği seçtiğimde başarı ciddi oranda düştü (0.833).
- Değerlendirme: PCA kullanımı, modelin eğitim hızını artırsa da %5'lik varyans kaybı, fiyat tahminindeki hassas detayları (örneğin nadir bir donanım paketinin etkisi)

kaybetmemize neden oldu. Bu yüzden en yüksek başarıyı tüm özelliklerin kullanıldığı ham (scaled) veri ile elde ettim.

3. Kümeleme (Segmentasyon) Analizi:

- Bulgu: DBSCAN algoritması ile veriyi analiz ettiğimde, araçların doğal olarak 3 ana segmente (küme) ayrıldığını gördüm. Ayrıca model, 29 adet aracı "gürültü" (outlier) olarak işaretledi.
- Yorum: Bu 3 küme muhtemelen "Ekonomik Şehir Araçları", "Orta Segment Aile Araçları" ve "Yüksek Performans/Lüks Araçlar" olarak ayrışıyor. Aykırı çıkan 29 araç ise ya veri giriş hatası olanlar ya da özel koleksiyonluk araçlar olabilir. Bu bilgi, pazarlama stratejisi için fiyat tahmininden bile değerli olabilir.

Yaptığım deneyler sonucunda, bu veri seti için en uygun modelin Decision Tree Regressor olduğu, veriyi en iyi temsil eden yapının ise boyut indirgeme yapılmadan tüm özelliklerin (One-Hot Encoded) kullanıldığı senaryo (R2 ~ 0.90) olduğu tespit edilmiştir. Grid Search ile yaptığım hiperparametre optimizasyonu (Max Depth: 15) aşırı öğrenmeyi engellemek için dengeli bir yapı kursa da, saf performans açısından temel model oldukça tatmin edici sonuçlar vermiştir.

Yöntem Model	1. Tüm Özellikler (Scaled)	2. SelectKBest (10 Özellik)	3. PCA (İndirgenmiş)
Bayesian Ridge	0.8675	0.7546	0.7946
Decision Tree	0.8972	0.8333	0.8761

	Model	Yöntem	R2 Score	MAE (Hata)	Özellik Sayısı
0	Decision Tree	1. Tüm Özellikler (Scaled)	0.897239	1907.356593	202
2	Decision Tree	3. PCA (İndirgenmiş)	0.876126	2024.951868	173
3	Bayesian Ridge	1. Tüm Özellikler (Scaled)	0.867499	2294.246005	202
1	Decision Tree	2. SelectKBest (10 Özellik)	0.833338	2384.847213	10
5	Bayesian Ridge	3. PCA (İndirgenmiş)	0.794551	2911.229283	173
4	Bayesian Ridge	2. SelectKBest (10 Özellik)	0.754571	3205.366119	10

E. Nurettin Kaplan - ElasticNet Regresyonu ve SGDRegressor Analizi

Veri setindeki fiyat dengesizliğini yönetmek, gereksiz karmaşıklığı önlemek ve pazar yapısını anlamak amacıyla aşağıdaki üç temel modelleme yaklaşımını seçtim:

1. ElasticNet Regresyon (Düzenlileştirilmiş Doğrusal Model):

- Seçim Nedeni: Veri setinde çok sayıda özellik olduğu için L1 (Lasso) ve L2 (Ridge) cezalandırma yöntemlerini bir arada kullanan bu modeli tercih ettim. Amacım hem gereksiz özellikleri baskılamak hem de aşırı öğrenmenin (overfitting) önüne geçmekti.
- Beklenti: Fiyat üzerindeki etkileri en dengeli şekilde modelleyerek, özellikle logaritmik dönüşüm yapılmış hedef değişkende en yüksek R2 skorunu vermesi hedeflenmiştir.

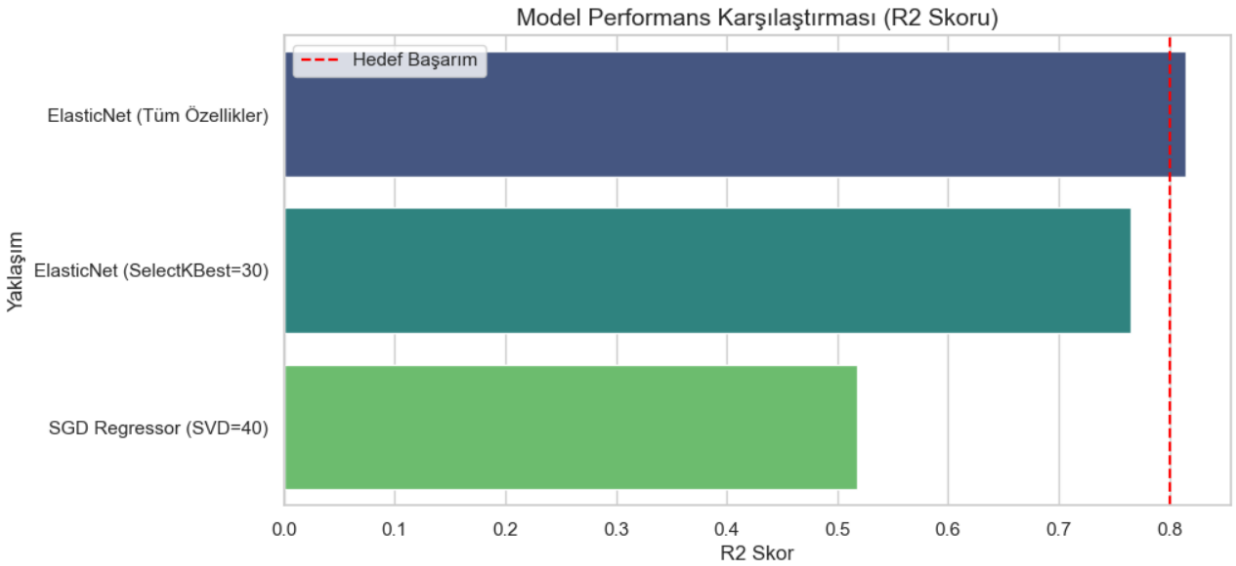
2. SGD Regressor - Stokastik Gradyan İnişi (Hız Odaklı Model):

- Seçim Nedeni: Büyük veri setlerinde optimizasyonun çok daha hızlı olması nedeniyle seçtim. Bu modeli, TruncatedSVD (Boyut İndirgeme) yöntemiyle birleştirerek seyrek matrislerdeki işlem yükünü hafifletmeyi amaçladım.
- Beklenti: Hız ve verimlilik sağlaması beklense de, boyut indirgeme işlemi sırasında bir miktar bilgi kaybı yaşanabileceği öngörülmüştür.

3. Gaussian Mixture Model - GMM (Olasılıksal Kümeleme):

- Seçim Nedeni: K-Means gibi katı sınırlar çizmek yerine, araçların fiyat ve performans özelliklerine göre hangi segmente (Ekonomik, Lüks vb.) ait olma olasılığının daha yüksek olduğunu görmek için kullandım.
- Beklenti: Veri setindeki doğal dağılımı yakalayarak araçları benzer gruplar altında toplamak ve pazarın yapısını çözümlemektir.

ElasticNet ve SGD modellerinin, özellik seçimi (Feature Selection) ve boyut indirgeme senaryolarındaki performans kıyaslaması şu şekildedir:



1. Model Karşılaştırması (Tüm Özellikler vs Seçilmiş Özellikler):

- Bulgu: En yüksek başarıyı, tüm özelliklerin kullanıldığı ElasticNet modeli sağlamıştır (R2: 0.814).
- Neden: Veri setindeki özelliklerin birçoğunun fiyat üzerinde anlamlı bir etkisi olduğu görüldü. SelectKBest ile özellik sayısı 30'a düşürüldüğünde performans 0.765'e geriledi; bu da elenen bazı özelliklerin model için hala bilgi taşıdığını gösteriyor.

2. Boyut İndirgeme ve Hız Odaklı Yaklaşım Etkisi:

- Bulgu: TruncatedSVD ile veriyi 40 bileşene indirgeyip SGD Regressor kullandığımda performans ciddi oranda düşmüştür (R2: 0.518).
- Değerlendirme: Bu düşüş, boyut indirgeme sırasında verideki varyansın (bilginin) önemli bir kısmının kaybolduğunu göstermektedir. Bu senaryoda hız kazanılsa da, doğruluktan verilen taviz çok yüksek olmuştur.

3. Veri Ön İşleme Etkisi (Log Dönüşümü):

- Bulgu: Başlangıçta sağa çarpık olan "Fiyat" verisine uyguladığım Log-Transform işlemi, regresyon modellerinin (özellikle ElasticNet'in) veriyi daha doğru öğrenmesini sağlamıştır.
- Yorum: Fiyat dağılımı normale yaklaştırılmasaydı, muhtemelen bu R2 skorlarına ulaşmak mümkün olmayacaktı.

	Model	Yöntem	R2 Score	MAE (Hata)	Özellik Sayısı
0	ElasticNet	1. Tüm Özellikler (Scaled)	0.814637	2154.898128	205
3	SGD Regressor	1. Tüm Özellikler (Scaled)	0.801522	2209.537623	205
1	ElasticNet	2. SelectKBest (30 Özellik)	0.765604	2469.012766	30
4	SGD Regressor	2. SelectKBest (30 Özellik)	0.759061	2503.737447	30
2	ElasticNet	3. SVD (40 Bileşen)	0.735601	2615.511510	40
5	SGD Regressor	3. SVD (40 Bileşen)	0.518220	3104.145595	40