**ARAÇ YAKIT TÜKETİMLERİNİN ÇOKLU REGRESYON İLE BELİRLENMESİ**

Kadir GÜRBÜZ

Fırat Üniversitesi Teknoloji Fakültesi Yazılım Mühendisliği Bölümü, ELAZIĞ 180541047@firat.edu.tr

**Özet**

*Yakıt tüketimini modelleme ve tahmin etme, araçların yakıt ekonomisini artırma ve filolarda dolandırıcılığı önlemek açısından önem taşımaktadır. Bir aracın yakıt tüketimi, yol koşulları, trafik ve hava durumu gibi dış faktörlerin yanı sıra mesafe, yük, araç özellikleri ve sürücü davranışı gibi birkaç faktöre de bağlıdır. Ancak, bu faktörlerin tümü ölçülemeyebilir veya yakıt tüketimi analizi için mevcut olmayabilir. Bu nedenle, zorluk, yakıt tüketimini yalnızca mevcut verilerle modellemek ve / veya tahmin ederken, diğer iç ve dış faktörlerden etkileri olduğu kadar dolaylı olarak da yakalamaktır. Model verilerdeki örüntüler öğrenilerek geliştirilebileceğinden, Makine Öğrenimi (ML) bu tür analizler için uygundur.* *Bu yazıda, otomobilin yakıt tüketimini topluluk yöntemi ile tahmin ettik. Topluluk yöntemi, yakıt tüketimini tahmin etmek için kullanılır ve gelişmiş tahmin sağlar LASSO Regresyonu, Elastik Ağ Regresyonu, Kernel Ridge Regresyonu, Regresyon Artırma ve meta modeller ve XGBoost kullandık. Ortalama hesaplamak için bu modelde tanıtılan makine öğrenimi modelinin öngördüğü değerler ve gerçek hedef değeri, yani otomobilin yakıt tüketimi tahmini için çoklu regresyonlar ve ortalamaları hesaplandı.*

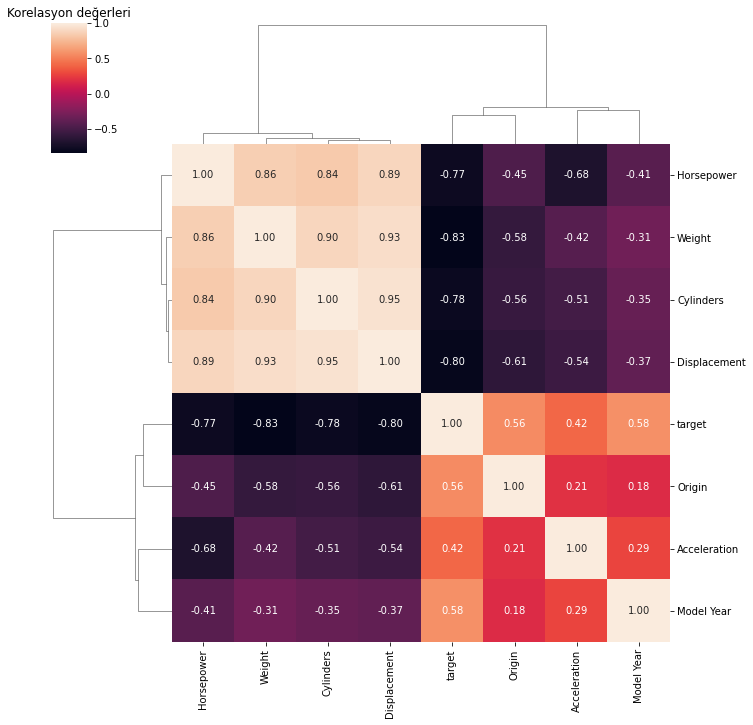
*Anahtar kelimeler: yakıt tüketim tahmini, regresyon teknikleri, toplu istifleme, yığılmış regrasyonlar, XGBoost, otomobil, tahmini performans, Filo, yakıt, trafik,*

**Giriş**

Yakıt tüketimleri doğrudan ve dolaylı olarak genel yaşam kalitesini, ev ekonomisini ve çevreyi, kentsel ulaşım sistemlerini derinden etkilenmektedir. Türkiye’de sera gazı emisyonu ya da Türkiye’de sera gazı salınımı kişi başına yaklaşık 6 tondur. Türkiye her yıl 500 milyon ton sera gazı salmaktadır. Bu oranla Türkiye, dünyanın yıllık salınımının yaklaşık olarak %1'ini meydana getirmektedir. Sera gazı salınımının yaklaşık üçte biri kömür kaynaklıdır. Çok büyük bir oranı ise araçlardan kaynaklı yakıt tüketimi sonucu yanan yakıtın atık gaz salınımından kaynaklanmaktadır.

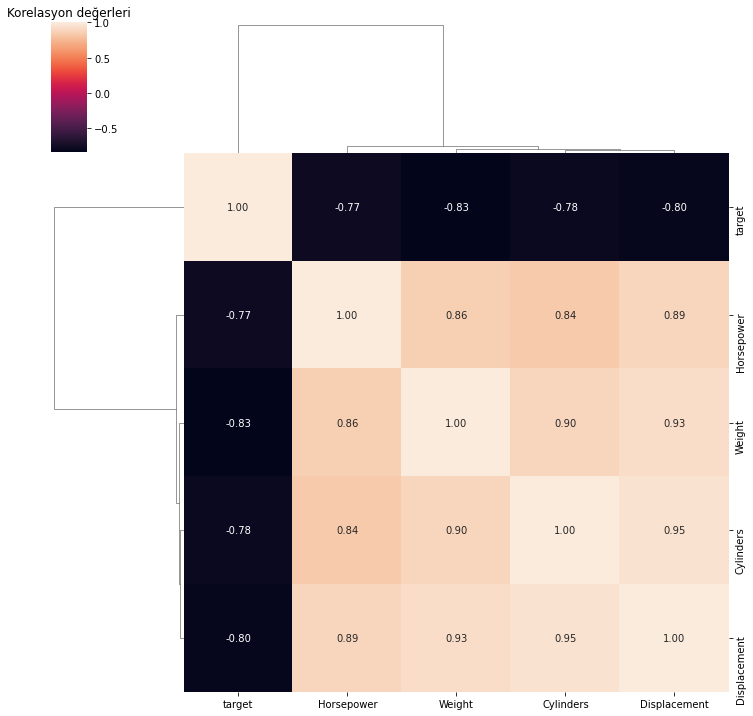
Motorlu taşıtların çalışmasında temel madde yakıttır. Ulaşım masraflarının büyük bir bölümünü yakıt oluşturmaktadır. Bu nedenle yakıt en yüksek ekonomi ile kullanılmalıdır. Otomobillerde yakıt ekonomisini etkileyen birçok faktör vardır. Bunlar taşıtın tasarımından , teknik durumuna , yol ve iklim şartlarından , sürücü deneyimine kadar çeşitli faktörleri içine alır . Ulaşım organizasyonu durumu da bunlara eklenebilir. Her geçen gün taşıtların performansı , sürüş kolaylığı ve konforu artırılmaya çalışılmakta bunları sağlarken de yakıt ekonomisini en üst düzeyde tutmak için taşıtın toplam verimini yükseltilmeye yönelik çalışmalarda yapılmaktadır.

İlk olarak veri setinden kayıp değerleri temizledik ve kayıp verilerin sonuçları etkilemesinin önüne geçtik. Bu temizleme işlemi aslında tüm kolonların ortalama değerlerini kayıp veriler yerine yazıp buradaki durumu bozmalarını engelleme şeklinde oldu.

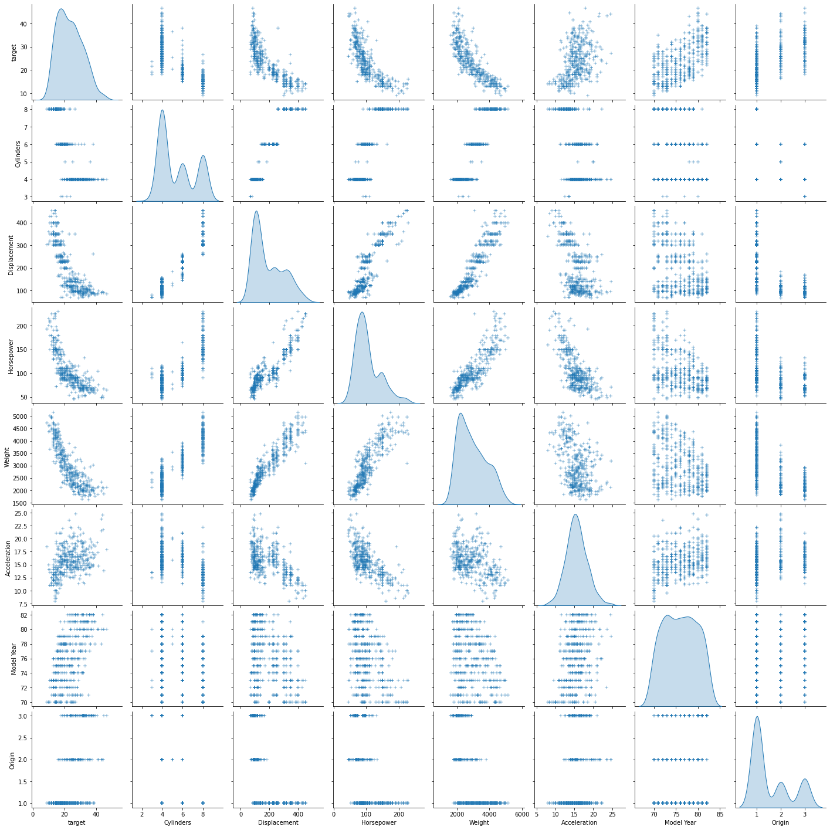
Kullandığımız veri setinde araç yakıt tüketimini etkileyen faktörleri belirleyebilmek için ilk olarak korelasyon değerlerini ölçümledik.

Korelasyon değerleri

Daha sonra bu korelasyon değerlerinden ilişki olarak en uygun olan kısmı seçtik. Bunu 0.75 değerinden daha büyük ilişkiye sahip olan verileri filtreleyerek gerçekleştirdik.



Filtrelenmiş korelasyon değerleri

Eş düzlemlilik grafiği ile birlikte hangi verilerin birbirleri arasında ne şekilde bir ilişki olduğunu grafiği yorumlayarak anlamamaya çalıştık. Daha sonraki uygulayacağımız işlemleri etkileyecek ve burada çıkan sonuçlara göre bir yol izlenecektir.

Eş düzlemlilik grafikleri

Bu grafikten hareketle Silindir ile Menşei kısımlarını kategorik, Displacement(İvme), Horsepower(Beygir), Weight(Ağırlık) arasında artmaları halinde yakıt tüketimi de artması durumu var diye bir yorum yapabiliriz.

Ayrıca

Silindir sayısına göre araç sayıları

Silindir-Araç sayısı

4 - 204

8 - 103

6 - 84

3 - 4

5 - 3

Menşei ye göre araç sayıları

Menşei-Araç sayısı

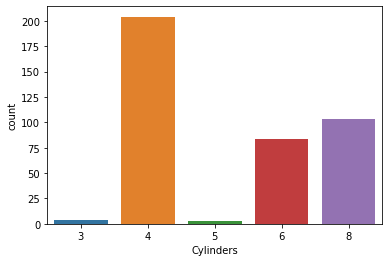
1 - 249

3 - 79

2 - 70

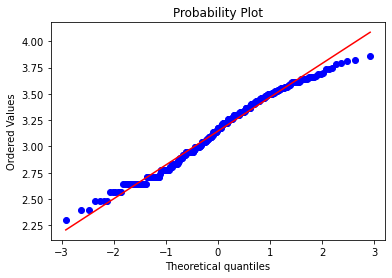
Hesaplamalara göre veri setinin mü ve sigma değerleri ise

mu: 23.472405063291134, sigma: 7.756119546409932

Silindir sayısına göre araç dağılım grafiği şöyledir.

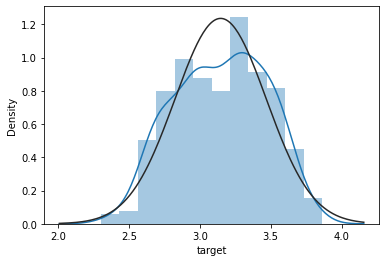
Silindir sayısına göre araç dağılımı

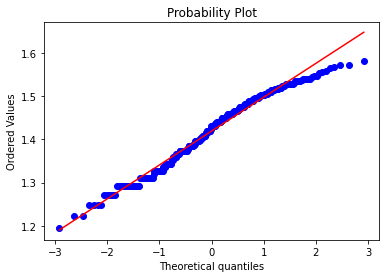
Aykırı değerlere göre doğrusal bir çizgi üzerinden veri seti verilerinin dağılımının grafiği :



Teorik değerler ve gerçek değerler karşılaştırması

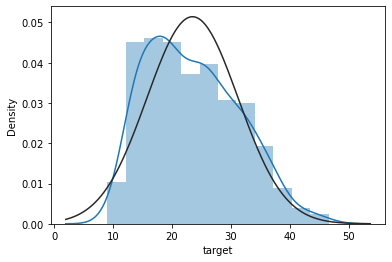
Aykırı değerleri ve çarpıklığı gidermek için gauss dağılımı uygulanmış veri setinin yeni grafiği ve normal grafik ile karşılaştırması:





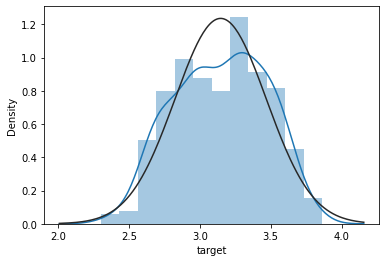
Gauss uygulanmış veri seti çarpıklığı

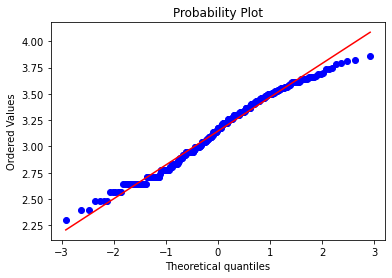
Gauss uygulanmış veri seti çizgisi

Target için de aynı çarpıklığı gidermek ve işlenmeye uygun hale getirmek gerekiyor.

Henüz gauss uygulanmamış target veri setinin çarpıklık görünümü

Burada çarpıklık sola doğru ve kuyruk sağa doğru olduğundan pozitif bir çarpıklık vardır diyebiliriz.

Target veri setine gauss uygulandıktan sonra ise:



Gauss uygulanmış target veri setinin çarpıklık görünümü

Gauss uygulanmış target veri setinin teorik-gerçek karşılaştırması

Daha sonra silindir sayısı ve menşei bölümünü kategorik türe uygun hale getiririz ve kategorik olarak sıralamak için tek bir hata türü uygularız. Birden çok hata türü veri setini yanlış olarak değerlendirmemize neden olur.

X\_train = X veri seti

X\_test = X test verisi

Y\_train = Target veri seti

Y\_test = Target eğitim veri seti

Sonrasında verinin analizi aşamasına geçiyor ve artık çeşitli analizler ile veri setini en az hatalı şekilde analiz etmeye çalışıyoruz.

**Regrasyon**

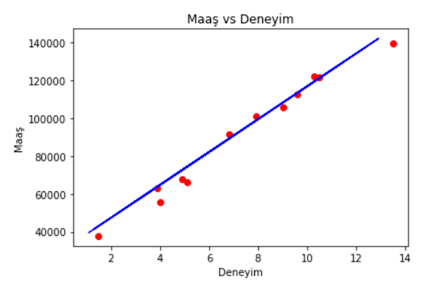
Regresyon Regresyon bir bağımlı değişken ile diğer birkaç bağımsız değişken arasındaki ilişkiyi belirler. Regresyon analizi, bağımsız değişkenlerin bazıları değiştiğinde bağımlı değişkenin nasıl değiştiğini anlamaya yardımcı olmaktadır. Bu yöntem değişkenler arasındaki neden sonuç ilişkisini tahmin etmek ve bulmak için kullanılır. Regresyon teknikleri çoğunlukla bağımsız değişkenlerin sayısına ve bağımsız ve bağımlı değişkenler arasındaki ilişkinin türüne göre farklılık gösterir ve tahmin yapmak için çeşitli regresyon teknikleri mevcuttur.

**Lineer Regrasyon**

Doğrusal korelasyon ve basit doğrusal regresyon, iki değişken arasındaki doğrusal ilişkiyi inceleyen istatistiksel yöntemlerdir. Burada şu farklılığı vurgulamakta fayda var: Korelasyon, iki değişkenin ne kadar ilişkili olduğunu gösterirken, doğrusal regresyon, iki değişken arasındaki ilişkiye dayanarak birinin değerini diğerinden tahmin etmeyi sağlayan bir denklem (model) oluşturmayı içerir. Doğrusal regresyon, bir dizi noktaya en uygun düz çizgiyi veya hiper düzlemi bulmak için kullanılmaktadır. Bir diğer ifadeyle doğrusal regresyon, en uygun düz çizgi (regresyon çizgisi olarak da bilinir ) kullanarak bağımlı değişken (Y) ile bir veya daha fazla bağımsız değişken (X) arasında bir ilişki kurar. Aşağıdaki grafikte kırmızı çizgi en uygun düz çizgi olarak adlandırılır.

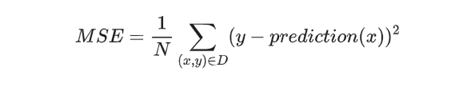
https://miro.medium.com/max/264/1*3n_BKBPw0xeNlQFz91X3Mg.pngBu model yalnızca bir özellik kullanıyor olsa da, daha karmaşık bir model, her biri ayrı bir ağırlığa sahip olan birden fazla özelliğe güvenebilir (w1, w2, vb.). Örneğin, üç özelliğe dayanan bir model aşağıdaki gibi görünebilir:

İki boyutlu dağılımlar, bir değişkenin diğerini nasıl etkilediğinin anlaşılmasında yardımcı olmaktadır. Bu bir neden-sonuç bağımlılığı olabilir, örneğin yağmur miktarı (neden), tarımsal üretimde (etki) bir artışa yol açar. İki boyutlu dağılımı temsil etmek için kartezyen bir koordinat sistemi kullanırsak, dağılım diyagramıyla bilinen bir dizi nokta elde ederiz; bu analiz, şekilde gösterildiği gibi her iki değişken arasındaki ilişkiyi niteliksel olarak incelememize izin verir.



Gerçek değerden uzaklaştıkça yaşadığımız kayıp bir kare ile artar. Bir modeli eğitirken sadece bir örnekteki kaybı en aza indirmekle ilgilenmiyoruz, tüm veri setimizdeki kaybı en aza indirmeyi önemsiyoruz. Loss, kötü bir tahminin cezasıdır. Yani kayıp modelin tahmininin tek bir örnek üzerinde ne kadar kötü olduğunu gösteren bir sayıdır. Modelin tahmini mükemmelse kayıp sıfırdır. Bir modeli eğitmenin amacı tüm örneklerde ortalama olarak düşük kayıplı bir ağırlık bulmaktır.

**Mean Squared Error (MSE)**

Modelimizin belirli bir veri seti üzerindeki performansını değerlendirmek için model tahminlerinin gerçek sonuçlarla ne kadar örtüştüğünü ölçmemize yarayacak yöntemlere ihtiyaç duyarız. Yani, belirli bir gözlem için tahmin edilen değerin o gözlemin gerçek değerine ne kadar yakın olduğunu sayısallaştırmamız gerekiyor. Regresyon için, en yaygın olarak kullanılan ölçüm Ortalama Karesel Hata (Mean Squared Error-MSE)‘dır. Tüm veri kümesinde örnek başına ortalama kare kaybıdır. MSE’yi hesaplamak için tek tek örnekler için tüm kare kayıpları toplanarak örnek sayısını bölünür.

(x,y) bir örnek:

→x modelin tahminlerde bulunmak için kullandığı özellikler kümesidir

→y örneğin etiketidir

→D birçok etiketli örnek içeren bir veri kümesidir. (x,y) çiftleri.

→ N- içindeki örnek sayısı D.

Burada bizim veri setimizde

Lineer katsayılar: [-1.52652234e-01 -1.10059003e-01 -3.51567523e-02 -7.81023601e-02

7.76057988e-02 -2.71614700e-01 1.86827885e-01 -7.97972799e-17

5.74192760e-02 2.73675388e-02 -1.43431578e-02 -6.77254422e-02

8.20686000e-02]

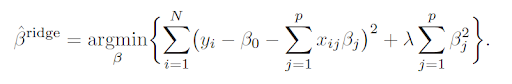
Doğrusal regrasyon ortalama mutlak hatası : 0.02098471106586962

Sonuçları elde edilmektedir.

**Ridge Regresyon (L2 regularization)**

Çok değişkenli regresyon verilerini analiz etmede kullanılır. Amaç hata kareler toplamını minimize eden katsayıları, bu katsayılara bir ceza uygulayarak bulmaktır. Over-fittinge karşı dirençlidir. Çok boyutluluğa çözüm sunar. Tüm değişkenler ile model kurar, ilgisiz değişkenleri çıkarmaz sadece katsayılarını sıfıra yaklaştırır. Modeli kurarken alpha (ceza) için iyi bir değer bulmak gerekir.

Burada Y, (n x 1) boyutlu bağımlı değişken, X, (n x p) boyutlu bağımsız değişkenler matrisi, β bilinmeyen katsayılar vektörü ve ε (n x 1) boyutlu hata terimi vektörüdür. Hataların dağılımı 0 ortalama ve 𝜎 2 𝐼𝑛 varyans ile normaldir.

Regresyon modelinde tüm değişkenler standartlaştırıldığında X'X matrisi korelasyon matrisi formundadır. Bu modelde β’nın EKK tahmin edicisi aşağıdaki gibidir:

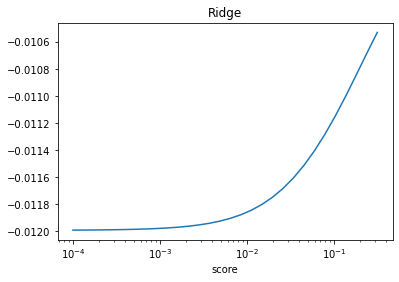
Bizim veri setimize göre

Sırt(Ridge) regrasyonu katsayısı : [-0.07617499 -0.10434789 -0.0756786 -0.06362033 0.08849698 -0.17538752

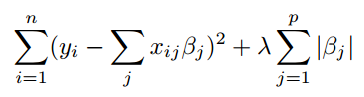
0.17840625 0. 0.02717488 -0.03019362 -0.02334111 -0.05192496

0.07526607]

Sırt(Ridge) regrasyonu en iyi tahmini : Ridge(alpha=0.31622776601683794, max\_iter=10000, random\_state=42)

Sırt(Ridge) Ortalama karesel hata : 0.018839299330570537

**Lasso Regresyon (L1 regularization)**

Ürettiği modelin tahmin doğruluğunu ve yorumlanabilirliğini arttırmak için hem değişken seçimi hem de regularization yapar. Aynı ridge regresyonda olduğu gibi amaç hata kareler toplamını minimize eden katsayıları, katsayılara ceza uygularayarak bulmaktır. Fakat ridge regresyondan farklı olarak ilgisiz değişkenlerin katsayılarını sıfıra eşitler.

Bizim veri setimize göre

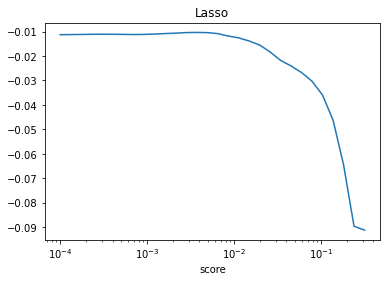
Lasso regrasyonu katsayısı : [-0.01692687 -0.10976505 -0.11721736 -0.03064576 0.09866154 -0.01243765

0.16495225 0. 0.00378698 -0.00505995 -0. -0.

0.07376033]

Lasso regrasyonu en iyi tahmini : Lasso(alpha=0.0037065129109221566, max\_iter=10000, random\_state=42)

Lasso Ortalama karesel hata : 0.016597127172690823



**Elastic Net**

Amaç ridge ve lasso regresyon ile aynıdır ama elastic net, ridge ve lasso regresyonu birleştirir. Ridge regresyon tarzı cezalandırma ve lasso regresyon tarzında değişken seçimi yapar.









Bizim veri setimize göre

ElasticNet katsayısı : [-0.0518437 -0.10923982 -0.09538543 -0.03946691 0.09541227 -0.07791667

0.14374215 0. 0. -0.01852327 -0. -0.00200172

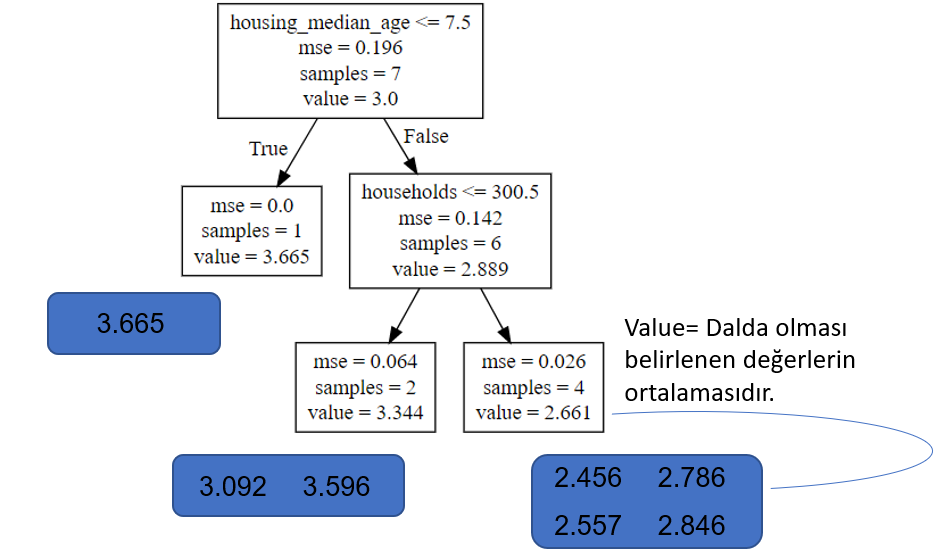
0.0793579 ]

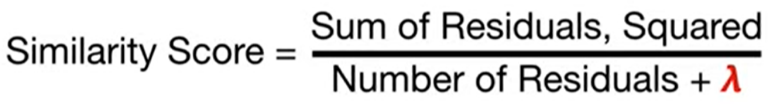
ElasticNet en iyi tahmin : ElasticNet(alpha=0.014873521072935119, l1\_ratio=0.15000000000000002,

max\_iter=10000, random\_state=42)

ElasticNet ortalama karesel değeri : 0.017234676963922283

**XGBoost Regrasyon Ağacı**

XGBoost(eXtreme Gradient Boosting), Gradient Boosting algoritmasının çeşitli düzenlemeler ile optimize edilmiş yüksek performanslı halidir. Tianqi Chen ve Carlos Guestrin’in 2016 yılında yayınladıkları “XGBoost: A Scalable Tree Boosting System” adlı makale ile hayatımıza dahil olmuştur. Algortimanın en önemli özellikleri yüksek tahmin gücü elde edebilmesi, aşırı öğrenmenin önüne geçebilmesi, boş verileri yönetebilmesi ve bunları hızlı yapabilmesidir. Tianqi’ye göre XGBoost diğer popüler algoritmalardan 10 kat daha hızlı çalışmaktadır. Daha az kaynak kullanarak üstün sonuçlar elde etmek için yazılım ve donanım optimizasyon tekniklerini uygulanmıştır. Karar ağacı tabanlı algoritmaların en iyisi olarak gösterilir.

Oluşturulan ağacın her bir dalı için benzerlik skoru(similarity score) hesaplanır. Benzerlik skoru verilerin dallarda ne kadar iyi gruplandığını gösterir.

Formül-1: Benzerlik Skoru Formülü

Benzerlik skorları hesaplandıktan sonraki soru daha iyi bir tahmin yapılıp yapılamayacağıdır. (Formüldeki lamda(λ), regülarizasyon parametresidir. Bu konuya daha sonra değinilecektir.) Bu soruyu cevaplamak için olabilecek tüm olasılıklardaki ağaçlar kurulur. Hepsi için benzerlik skorları hesaplanır. Hangi ağacın daha iyi olduğuna karar vermek için kazanç(gain) hesaplanacaktır. “Benzerlik skoru” ile dallar değerlendirilirken, “kazanç” ile bütün ağaç değerlendirilmektedir. Kazanç = Sol Benzerlik Skoru+ Sağ Benzerlik Skoru – Önceki Ağacın Benzerlik Skoru ‘dur.

Bizim veri setimize göre

XGB regrasyon ağacı ortalama karesel hatası : 0.01932020789212713

Hata oranı fazla olduğundan parametre düzenlemesi ile tekrarlanırsa

XGB regrasyon ağacı ortalama karesel hatası : 0.017444718427058307

**En iyi modellerin ortalaması ile tahmin**

Bu en etkili yöntemdir. Bulunan en iyi sonuçların yöntemlerinin sonuçları ortalaması yöntemi ile bulunur.

Bizim veri setimize göre

En iyi modellerin ortalaması ile ortalama karesel hata : 0.017415759476209068

**Son analizler**

**[StandartScaler] ile elde edilen sonuçlar**

Doğrusal regrasyon sonucuna göre hata oranı : 0.020632204780133088

Sırt(ridge) regrasyon sonucuna göre hata oranı : 0.01972533801080122

Lasso regrasyon sonucuna göre hata oranı : 0.017521594770822504

ElasticNet regrasyon sonucuna göre hata oranı : 0.01749609249317252

**[RobustScaler] ile elde edilen sonuçlar (Aykırı değerleri veriden uzaklaştırır)**

Doğrusal regrasyon sonucuna göre hata oranı : 0.02098471106586962

Sırt(ridge) regrasyon sonucuna göre hata oranı : 0.018839299330570537

Lasso regrasyon sonucuna göre hata oranı : 0.016597127172690823

ElasticNet regrasyon sonucuna göre hata oranı : 0.017234676963922283

XGBoost parametresiz regrasyon sonucuna göre hata oranı : 0.01932020789212713

XGBoost parametreli regrasyon sonucuna göre hata oranı : 0.017444718427058307

En iyi modellerin ortalaması sonucuna göre hata oranı : 0.017415759476209068

**StandartScaler** kullanıldığında yani aykırı veriler normal verilerden uzaktaştırılmadan veri seti

değerlendirildiğinde sonuçlar yukarıda ki gibidir ve en iyi başarımı ElasticNet ile elde ediyoruz.

**RobustScaler** kullanıldığında ise aykırı veriler veri setinden uzaklaştırılarak değerlendirme yapılır.

Bununla birlikte Doğrusal regrasyonun başarımı düşerken diğerlerinde az bir iyileşme de olsa en iyi

başarımı Lasso regrasyonu sağlamaktadır. Lasso regrasyonunun başarımı ciddi oranda artmıştır ve diğer

tüm sonuçlardan daha iyi bir başarıma sahip olmuştur.

# Kaynakça

*Elastic net regularization*. (2021, Mart 19). Wikipedia: https://en.wikipedia.org/wiki/Elastic\_net\_regularization adresinden alındı

Moradi, E., & Monero, L. M. (2020, 10 14). *Transportation Research Part D: Transport and Environment*. Sciencedirect: https://www.sciencedirect.com/science/article/abs/pii/S136192092030763X adresinden alındı

Muratlar, E. R. (2020, Mart 24). *XGBoost Nasıl Çalışır? Neden İyi Performans Gösterir? - Veri Bilimi Okulu*. Veri Bilimi Okulu: https://www.veribilimiokulu.com/xgboost-nasil-calisir/ adresinden alındı

Oleszak, M. (2019, Kasım 12). *(Tutorial) Regularization: Ridge, Lasso and Elastic Net - DataCamp*. Datacamp: https://www.datacamp.com/community/tutorials/tutorial-ridge-lasso-elastic-net adresinden alındı

Qin, W., Xu, Y., Ping, P., Miyajima, C., & Takeda, K. (2019, 6 3). *Impact of Driver Behavior on Fuel Consumption: Classification, Evaluation and Prediction Using Machine Learning*. Ieee: https://ieeexplore.ieee.org/abstract/document/8727915 adresinden alındı

Rahman, A., & Smith, A. D. (2017, Haziran 21). *Predicting fuel consumption for commercial buildings with machine learning algorithms*. ScienceDirect: https://www.sciencedirect.com/science/article/abs/pii/S0378778817322922 adresinden alındı

Taş, B. (2020, Mayıs 15). *Doğrusal Regresyon (Linear Regression) | by Berna Taş | Medium*. Medium: https://bernatas.medium.com/do%C4%9Frusal-regresyon-linear-regression-8f562c19aadf adresinden alındı

*Türkiye’de sera gazı emisyonu - Vikipedi*. (2021, Mayıs 26). Vikipedi: https://tr.wikipedia.org/wiki/T%C3%BCrkiye%E2%80%99de\_sera\_gaz%C4%B1\_emisyonu adresinden alındı

*UCI Machine Learning Repository: Auto MPG Data Set*. (1993, Temmuz 07). UCI : https://archive.ics.uci.edu/ml/datasets/auto+mpg adresinden alındı

Uyanık, T., Arslanoğlu, Y., & Karatuğ, Ç. (2020, Mayıs 21). *Machine learning approach to ship fuel consumption: A case of container vessel*. ScienceDirect : https://www.sciencedirect.com/science/article/abs/pii/S1361920920305769 adresinden alındı