BIL441 Bike Sharing Demand Projesi

Kaan Canbolat
TOBB ETÜ BİLGİSAYAR MÜHENDİSLİĞİ
Ankara,Türkiye

***Abstract*—Bu raporda BİL470 Machine Learning dersinin projesi için bike sharing demand konusunda öznitelik analiz edilmesi, çıkarımı ve 5 farklı model üzerinde verilerin tahminlere göre verdiği sonuçlar incelenmiştir.**

I. GIRIŞ

A. Motivasyon

Gelişen dünyada düşük karbon hareketinin hızla ilerlemesi, insan nüfüsunun artmasıyla birlikte artan araç sayısı trafik sıkışıklığı yaratmakta ve kalabalık şehirlerdeki yaşanabilirlik seviyesini azaltmaktadır. Bunun çözümü olarak insanları toplu taşıma araçlarına sıfır karbon tüketimine sahip elektrikli araçlara ve bisikletlere yönelim gün geçtikte artmaktadır. Gerçek zamanlı olarak bisiklet kiralama konusunda halkın ihtiyaçlarını yönlendirmek için bisiklet kiralama sayısını doğru tahmin edebilmek, firmaların kiralama politikası geliştirmesine elverişlidir.

Bu projede, bisiklet kiralama verilerinin tarihsel verilerine dayanarak, Kaggle üzerinden temin edilen veri seti ile verilerin özelliklerine göre, bisiklet kiralama talebini tahmin etmek aynı zamanda günümüzde sayısı çok artan elektrikli scooter kiralama oranıyla benzerlik yaratılabilir ve scooterleri ihtiyaca göre konumlandırarak verimliliği artırabiliriz.

B. Problem Tanımı

Bisiklet talep tahmini bir zaman serisi regresyon problemidir.

C. Amaç / Hedef

Bu projenin amacı, bulunan saat içerisindeki özniteliklerin durumuna göre talep edilen bisiklet sayısını eldeki verileri işleyerek doğru model ile birlikte doğru tahmin edebilme mekanizması oluşturabilmek. Farklı modellerle elde edilen sonuçları karşılaştırmak ve analiz etmektir.

D. Başarım Metrikleri

Kullandığım metrikler, MSE (Mean Squared Error), RMSE (Root Mean Squared Error), MAE ˇ (Mean Absolute Error), ve R2 Score. Kullandığım metrikler ile birlikte başarılı tahmin yaptığını R2 Score'unun %90'ın üzerinde olmasıyla birlikte kabul ederek ilerleyeceğim.

II. LITARATÜR ARAŞTIRMASI

Literatür taramasında genellikle turizm alanında gelecek turist sayısını tahmin etmek amacıyla yapılan araştırmalar ile bağdaştırdım. Çünkü hava durumu, ay, saat, günün tatil olup olmadığı gibi bilgiler ile ortak paydada çalıştığı çıkarımında bulundum.[1][2] Elde ettiğim bilgilere göre kullanılan modeller ARIMA, Random Forest, Gardient boosting , Linear Regression, Arma, KELM, LSSVR-GSA(least squares support vector regression with gravitational search algorithm), Lasso, Decision Tree gibi modeller kullanıldığını gördüm.

III. VERI SETI, VERI ÖZELLIKLERI VE ÖZNITELIKLERI

A. Veri Kaynağı

https://www.kaggle.com/competitions/bike-sharing-demand/data

B. Veri Kümesi

- Tarih(gün,ay,yıl,saat)
- Mevsim
- Günün ulusal tatil olup olmadığı
- Günün çalışma günü olup olmadığı
- Hava Durumu(1-4 arasında; elverişliden zorluya gidecek şekilde)
- Sıcaklık Derecesi
- Hissedilen Sıcaklık
- Bağıl Nem
- Rüzgar Hızı
- Kayıtlı kullanıcı sayısı
- Kayıtsız kullanıcı sayısı
- Toplam kullanıcı sayısı

Toplam 10886 adet veri vardır.

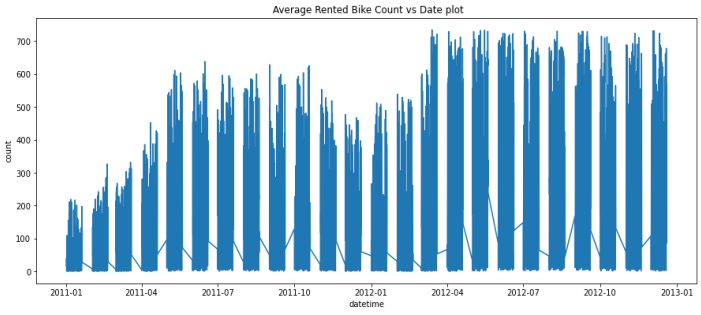
C. Öznitelik açıklamaları

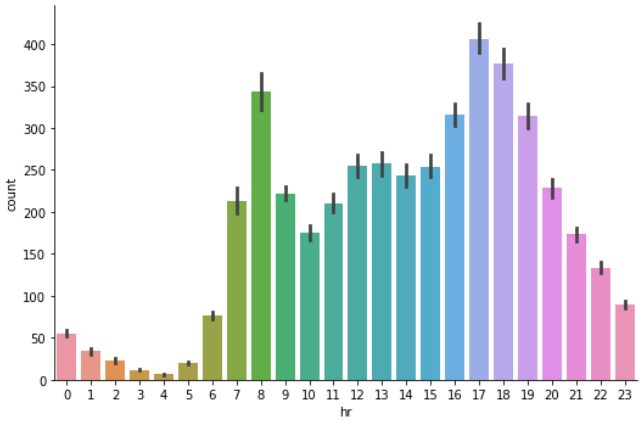
Veri kümesinde eksik veya hatalı veri içeren bir öznitelik bulunmuyor ve structured bir yapıdadır.

D. Sıralama, Kategorizasyon

Veriler tarihe göre sıralıdır ve her saatin verisi bulunmaktadır. Saate göre karşılaştırılabilir.

E. Verinin dağılımı





F. *Öznitelik Seçimi veya Düzenlemesi*

Tarih formatı yıl:ay:gün:saat şeklindeydi hepsini ayırdım. Yıl değişkenine one hotcode uyguladım. Saati de yukarıdaki dağılıma göre 7-9 arası ve 16-19 arasını 'a', 9-16 arasını 'b', 19-7 arasını 'c' şeklinde grupladım sonrada one hotcode uyguladım. Rüzgar hızı ve nem oranına da gruplama yaparak one hotcode uyguladım. Çünkü diğer özniteliklerin verilerine göre büyük fark oluşturuyordu diğer bir yöntem olarak normalizasyon yapabilirdim. Hava durumu 4 farklı şekilde gösteriliyordu. Hava durumunu ve mevsime'de one hotcode uyguladım.

Kayıtlı ve normal kullanıcıları toplam kullanıcı özniteliğinde birleştridim.

G. *Öznitelikler Arasındaki İlişkiler*

Hissedilen sıcaklık ve normal sıcaklık arasında yüksek koralasyon vardı. Fakat kaldımadan önceden moddeller içerisinde önemine baktığımda bazı modellerde öneminin yüksek olduğunu gözlemlediğimde kaldırmaktan vazgeçtim.

IV. KULLANILAN MODELLER

Verim sıralı fakat değerlendirildiği durumu düşündüğümüzde bir gün hava sıcak iken diğer gün havanın durumu tamamen farklı olabiliyor Mevsim özniteliği adı altında grafikte incelediğimizde ortalama verinin değiştiğini gözlemleyebiliyoruz. O yüzden sıralama etkisini yitiriyor. Verimin %80 eğitim %20 si test olacak şekilde ayrım yaptım.

5 farklı model kullandım bunlar: Linear Regression, Decision Tree Regressor, Gradien Boosting Regressor, Random Forest Regressor, XGBRegressor.

A. *Linear Regression Model*

Doğrusal regresyon analizi, bir değişkenin değerini başka bir değişkenin değerine göre tahmin etmek için kullanılır. Tahmin etmek istediğiniz değişkene bağımlı değişken denir. Diğer değişkenin değerini tahmin etmek için kullandığınız değişkene bağımsız değişken denir. Bu analiz biçimi, bağımlı değişkenin değerini en iyi tahmin eden bir veya daha fazla bağımsız değişkeni içeren doğrusal denklemin katsayılarını tahmin eder. Doğrusal regresyon, tahmin edilen ve gerçek çıktı değerleri arasındaki tutarsızlıkları en aza indiren düz bir çizgiye veya yüzeye uyar. Bir dizi eşleştirilmiş veri için en uygun doğruyu bulmak için "en küçük kareler" yöntemini kullanan basit doğrusal regresyon hesaplayıcıları vardır. Daha sonra X'in (bağımlı değişken) değerini Y'den (bağımsız değişken) tahmin edersiniz.[5]

B. *Decision Tree Regressor*

Karar ağacı, bir ağaç yapısı şeklinde regresyon veya sınıflandırma modelleri oluşturur. Bir veri kümesini giderek daha küçük alt kümelere ayırırken aynı zamanda ilişkili bir karar ağacı aşamalı olarak geliştirilir. Nihai sonuç, karar düğümleri ve yaprak düğümleri olan bir ağaçtır. Bir karar düğümünün (örneğin, Outlook) her biri test edilen öznitelik için değerleri temsil eden iki veya daha fazla dalı (örneğin, Güneşli, Bulutlu ve Yağmurlu) vardır. Yaprak düğümü (örn. Oynanan Saat) sayısal hedefle ilgili bir kararı temsil eder. Kök düğüm adı verilen en iyi tahmin ediciye karşılık gelen bir ağaçtaki en üstteki karar düğümü. Karar ağaçları hem kategorik hem de sayısal verileri işleyebilir.[4]

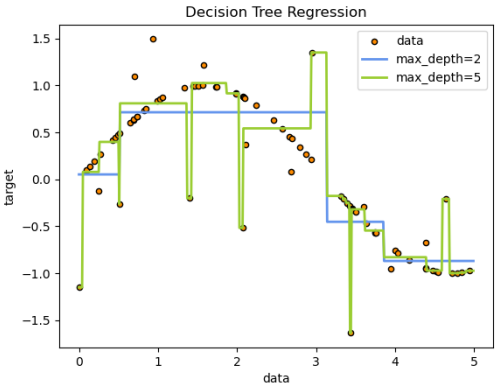


Figure [3]

C. *Graident Boosting Regressor*

Gradyan artırma, bir yükseltme modelidir. Artırma modellerinin önemli noktası önceki hatalardan ders almaktır. Gradyan artırıcı karar ağaçlarında, güçlü bir öğrenici bulmak için birçok zayıf öğreniciyi birleştiriyoruz. Buradaki zayıf öğreniciler, bireysel karar ağaçlarıdır.

Tüm ağaçlar seri olarak bağlanır ve her ağaç bir önceki ağacın hatasını en aza indirmeye çalışır.

Bu sıralı bağlantı nedeniyle, yükseltme algoritmalarının öğrenilmesi genellikle yavaştır (öğrenme hızı parametresi kullanılarak geliştirici tarafından kontrol edilebilir), ancak aynı zamanda oldukça doğrudur. İstatistiksel öğrenmede, yavaş öğrenen modeller daha iyi performans gösterir.

D. *Random Forest Regressor*

Random Forest, regresyon ve sınıflandırma dahil olmak üzere çeşitli görevler için kullanılabilen sağlam bir makine öğrenimi algoritmasıdır. Bu bir topluluk yöntemidir, yani rastgele bir orman modeli, tahminciler adı verilen ve her biri kendi tahminlerini üreten çok sayıda küçük karar ağacından oluşur. Rastgele orman modeli, daha doğru bir tahmin üretmek için tahmincilerin tahminlerini birleştirir. [6]

E. *XGBRegressor*

XGBoost, aşırı gradyan artırma anlamına gelen bir yazılım kitaplığıdır. Kütüphane, Tianqi Chen tarafından bir araştırma projesi olarak oluşturulan ve o zamandan beri diğer geliştiriciler tarafından katkıda bulunulan gradyan artırma makinelerinin bir uygulamasıdır. XGBoost,

hesaplama hızına ve model performansına odaklanır ve Gradient Boost, Stokastik Gradient Boost ve Düzenli Gradient Boost gibi optimizasyon özellikleri sunar.

Kaynak İşlem süresi ve bellek kaynaklarının verimliliğini en üst düzeye çıkarmak için XGBoost algoritması uygulandı. Algoritma, veri kümelerindeki eksik değerleri otomatik olarak işleyerek Seyrek Farkındalık uygulamasına sahiptir. Ek olarak, algoritma, önceden takılmış bir modelin yeni veriler üzerinde sürekli eğitimine izin verir.[7]

V. GIRIŞ

Bu bölümde seçtiğim modeler ile aldığım sonuçların doğruluklarını test ettiğim metrikler ile birlik sonuçlarını sunacağım, gerçek ve tahmin şeklinde grafiklerini, hangi modelde hangi özniteliğin daha etkiliği olduğunun grafiğini göstererek sonuçlarımı açıklamaya çalışacağım.

Modellerin optimal metrikler ile çalıştırılması için Grid Search cross validation methodunu kullanarak eğitime dahil ettim. Aynı zamanda hatayı azalmak için 10'ar kez modelleyip ortalamasını alarak sonuçları sundum.

A. *Linear Regression*

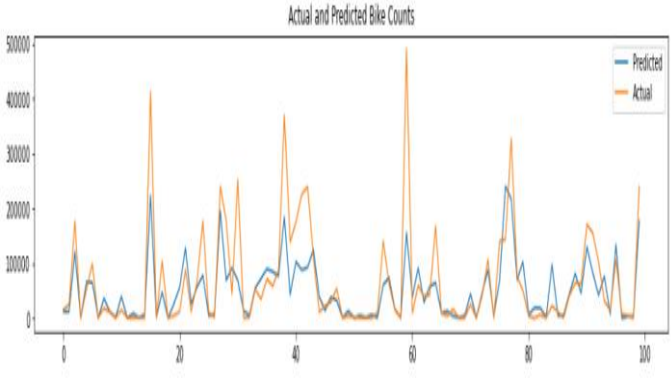
MSE : 4023994696.9458675

RMSE : 63434.96430948683

MAE : 34837.505104005606

Train R2 : 0.5341241211518531

Test R2 : 0.5232870310342002



Yukarıdaki grafikte sarı olan gerçek veri, mavi tahmin edilendir. sonuçlardan görüldüğü gibi linear regression modeli train skoru bile düşük olmasından verileri tamda doğru anlamlandıramadığını görüyoruz.

B. *XGBRegressor*

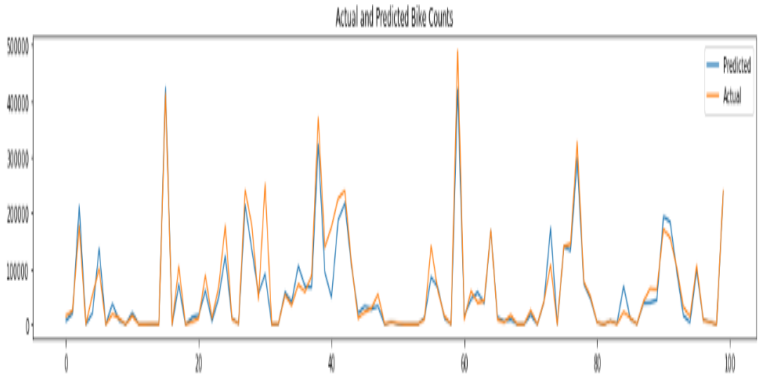
MSE : 677750553.4882952

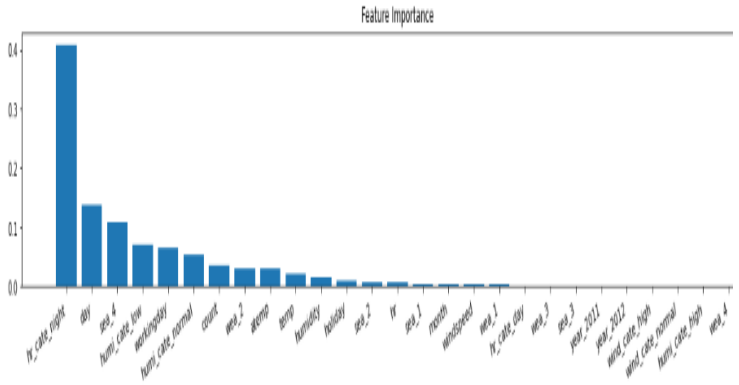
RMSE : 26033.64272414245

MAE : 12742.034899626447

Train R2 : 0.9862558991278003

Test R2 : 0.919708522772945





Grafikler büyük olduğu için açıklamayı altında yapma gereksiniminde bulundum. Öznitelik önemine baktığımızda bu modelde en önemli saat, gün, mevsim şeklinde ilerdiğini görüyoruz.

C. *Random Forest Regressor*

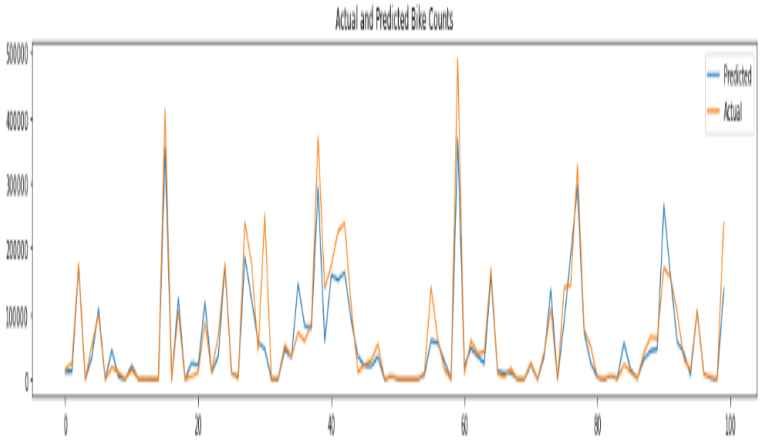
MSE : 1637598410.2253878

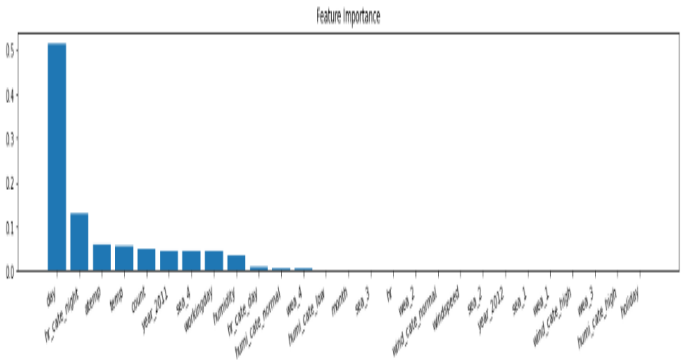
RMSE : 40467.25108313373

MAE : 20783.73950714847

Train R2 : 0.818501773695526

Test R2 : 0.8059976568297351





Öznitelik önem sırası; Gün, saat, hissedilen sıcaklık, normal sıcaklık şeklinde ilerlemektedir.

D. Gradient Boosting Regressor

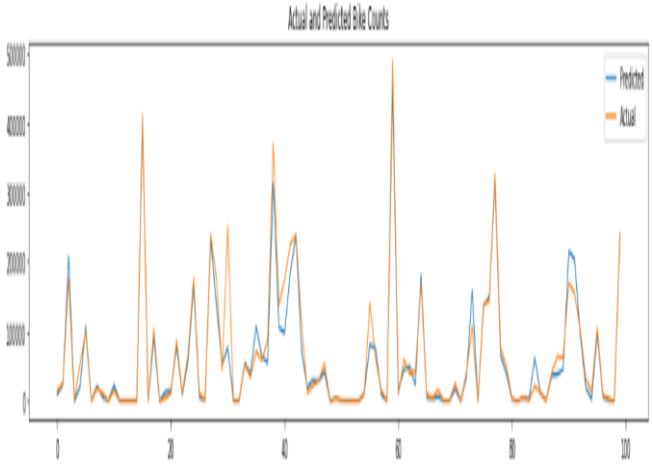
MSE : 760963910.9646767

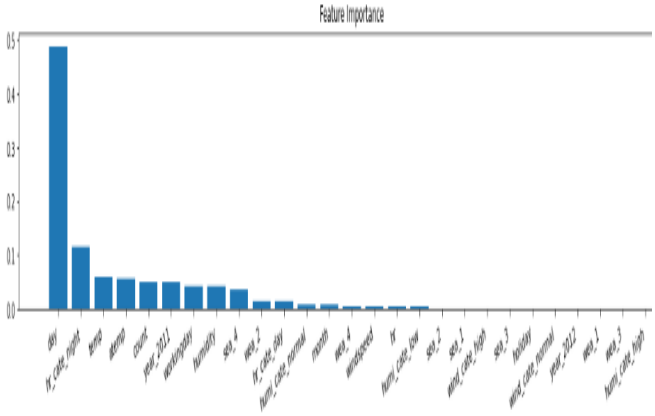
RMSE : 27585.57432725802

MAE : 13163.587066243832

Train R2 : 0.9607100904341709

Test R2 : 0.9098504365457722





Öznitelik önem sıralaması: Gün, saat, hissedilen sıcaklık, normal sıcaklık şeklinde ilerlemektedir.

E. Decision Tree Regressor

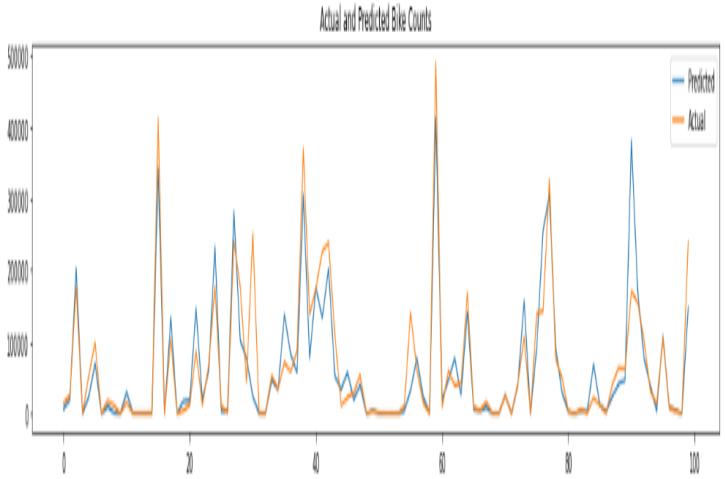
MSE : 1913481550.061737

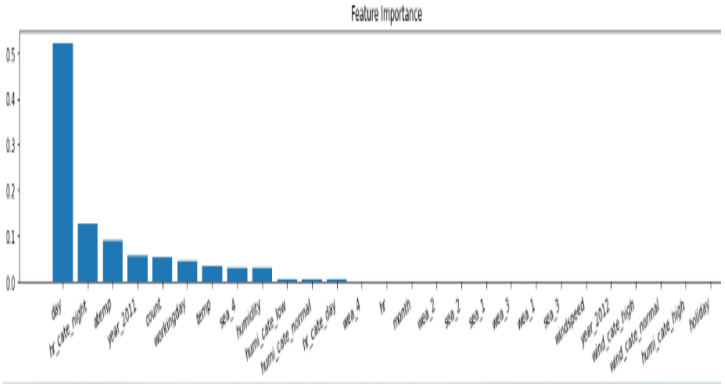
RMSE : 43743.360068263355

MAE : 22362.633404484608

Train R2 : 0.8128350039441503

Test R2 : 0.7733144451001541





Öznitelik önem sıralaması: Gün, saat, hissedilen sıcaklık, çalışma günü mü şeklinde ilerlemektedir.

VI. LITA

BIL470 dersi kapsamında yapmış olduğum Bike Sharing Demand projemde 5 farklı algoritma ile modeller oluşturdum ve en yakın saatte toplam bisiklet sayı tahmini yapan modeli elde etmeye çalıştım. Elde ettiğim sonuçları performans metrikleri ile kıyasladım.

Projenin sonucunda bir forecasting regression problemi olan bike sharing demand konusunda makine öğrenmesi ve doğru seçilmiş moddler ile gayet yakın tahminler yapılabileği görülüyor. Fakat sonuçta bu modeli biz ileri zaman için yapacağımız için o günün hava şartları da tahmine dayalı olacaktır. Bu yüzden modelimizin tahmininin doğruluğu aynı zamanda hava şartlarını tahmin eden sistemlerin doğruluğu ile paralel sonuçlar verecektir. Bu projede seçtiğim modelleri literatür araştırmasında elde ettiğim bilgiler doğrultusunda seçtim. Fakat linear regression modelinde isteğim sonucu elde edemedim. Özniteliklerin fazla olması uzayı büyüttüğü için linear regressionda uzayı doğru bölemediği için bu

oranda yanlış sonuç çıktı şeklinde yorumladım. İlerleyen süreçte yüksek modelleri bir ensamble network şeklinde kullanarak tahmin oranını artırabileceğimi düşünüyorum.

Seçilen modeller arasında en iyi sonuç veren modelimiz XGBRegressor'dur.

REFERENCES

[1] https://www.sciencedirect.com/science/article/pii/S0377221706012057

[2] https://www.mdpi.com/2071-1050/14/5/2564/htm

[3] https://scikit-learn.org/stable/auto_examples/tree/plot_tree_regression.html#sphx-glr-auto-examples-tree-plot-tree-regression-py

[4] https://www.saedsayad.com/decision_tree_reg.htm

[5] https://www.ibm.com/topics/linear-regression

[6] https://deepai.org/machine-learning-glossary-and-terms/random-forest

[7] https://deepai.org/machine-learning-glossary-and-terms/xgboost