**T.C**

**NECMETTİN ERBAKAN ÜNİVERSİTESİ**

**UYGULAMALI BİLİMLER FAKÜLTESİ**

**YÖNETİM BİLİŞİM SİSTEMLERİ BÖLÜMÜ**

**İKİNCİ ÖĞRETİM**

**MAKİNE ÖĞRENMESİ**

**FİNAL ÖDEVİ**

*ARAŞTIRMA KONUSU*

***KARAR AĞAÇLARI YÖNTEMİ İLE EV FİYAT TAHMİNİ***

**ÖDEV SAHİBİ**

**-----**

*ÖMER FARUK YÜCEL*

*21410082030*

1. **GİRİŞ**

Bu proje, ev fiyatlarını tahmin etmek amacıyla karar ağaçları yöntemini kullanarak bir regresyon modeli oluşturmayı hedeflemektedir. Ev fiyatları, birçok faktörün etkileşiminden kaynaklanan karmaşık bir konudur ve makine öğrenimi, bu tür ilişkileri öğrenerek doğru tahminler yapabilir.

Projede kullanılan makine öğrenimi modeli, **DecisionTreeRegressor** olarak adlandırılan bir karar ağacı regresyon modelidir. Karar ağaçları, ağaç yapısı içindeki karar düğümleri ve dallar arasındaki ilişkilerle, veri setindeki desenleri anlamak ve hedef değişkeni tahmin etmek için kullanılır.

Bu projede tercih edilen yöntemin neden karar ağaçları olduğu, ev fiyatları tahmininde bu yöntemin avantajları ve dezavantajları gibi konulara da değinilecektir. Karar ağaçlarının, modelin içsel mantığını anlama, açıklanabilirlik ve kolay yorumlanabilirlik gibi avantajlara sahip olduğu belirtilecektir.

Projede kullanılan modelin temel özellikleri ve avantajları, ev fiyatlarının tahmininde kullanılabilecek en iyi yöntem olduğunu destekleyecek şekilde vurgulanacaktır. Ayrıca, regresyon problemleri için neden bu modelin tercih edildiği ve projenin genel amacı projenin başlangıcında açıklanacaktır.

1. **DECISION TREE (KARAR AĞACI) ALGORİTMASI NEDİR?**

Karar ağacı, belirli bir soruna yönelik tüm potansiyel çözümleri haritalandıran akış şeması benzeri bir diyagramdır. Bu algoritma, bir karar ağacı şeklinde düzenlenmiş bir dizi soru ve kararlar içerir. Bu sorulara verilen cevaplara göre, algoritma adım adım bir çözüm bulur.

Örneğin, bir karar ağacı ile, bir hayvanın türünü belirlemeye çalıştığımızı düşünelim. İlk soru belki de "Bu hayvanın tüyleri var mı?" olabilir. Eğer cevap "Evet" ise, bir sonraki soru "Bu hayvan uçabilir mi?" olabilir. Bu şekilde, her soru ve cevap bir sonraki adıma yönlendirir ve nihayetinde hayvanın türünü belirleyen bir karara ulaşılır.

Karar ağaçları, basit ve anlaşılır olmalarıyla bilinir. Her bir adımda bir karar verilir ve bu adımların bir araya gelmesiyle daha karmaşık problemleri çözebilirler. Ayrıca, eldeki veri setine uyum sağlamak ve özellikle sınıflandırma görevlerinde etkili sonuçlar elde etmek için eğitilebilirler.

Karar ağaçları, sınıflandırma ve regresyon problemleri için kullanılan güçlü ve anlaşılabilir bir makine öğrenimi tekniğidir. Temel amacı, bir dizi karar ve koşul kullanarak veri setindeki desenleri öğrenmek ve bir hedef değişkeni tahmin etmektir. İşte karar ağaçlarının temel özellikleri:

**2.1. Ağaç Yapısı:**

* Karar ağaçlarının ilk hücrelerine kök denir. Her bir gözlem kökteki koşula göre “Evet” veya “Hayır” olarak sınıflandırılır.
* Kök hücrelerinin altında düğümler bulunur. Her bir gözlem düğümler yardımıyla sınıflandırılır. Düğüm sayısı arttıkça modelin karmaşıklığı da artar.
* Karar ağacının en altında yapraklar bulunur. Yapraklar, bize sonucu verir.

metin, ekran görüntüsü, diyagram, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

**2.2. Karar Süreci:**

Karar ağaçları, bir örnek verildiğinde, ağaç yapısını kullanarak kararlar alır. Her düğümde, belirli bir özellik değeri kontrol edilir ve bu değere göre kararlar alınır. Bu kararlar, ağacın dallarında ilerleyerek yapılır ve sonunda bir yaprak düğümüne ulaşılır. Yaprak düğümünde bulunan değer, hedef değişkenin tahminidir.

**2.3. Bölme Kriterleri:**

Karar ağaçları, veri setini bölerek en iyi tahminleri yapabilmek için özellikleri seçme stratejileri kullanır. Bu stratejiler, veri setini en homojen alt gruplara bölen kriterlere dayanır. İki temel bölme kriteri şunlardır:

* **Gini Impurity (Saflık):** Veri setinin homojenliğini ölçen bir kriterdir. Dallara ayrıldıkça gini impurity azalmaya çalışılır.
* **Entropy (Entropi):** Bilgi kazancını maksimize etmeye çalışan bir kriterdir. Daha homojen alt gruplar, daha düşük entropi değerine sahiptir.

**2.4. Aşırı Uyma (Overfitting) Kontrolü:**

Karar ağaçları, eğitim veri setine aşırı uyum yapma eğilimindedir. Bu nedenle, ağacın derinliği, minimum bölme örnek sayısı gibi hiperparametreler kullanılarak aşırı uyum kontrol edilir. Aşırı uyum kontrolü, modelin genelleme yeteneğini artırır.

**2.5. Avantajlar:**

1. **Açıklanabilirlik:** Karar ağaçları, modelin nasıl kararlar aldığını açıklamak açısından oldukça anlaşılabilirler. Her düğümdeki kararlar ve sınıflar açıkça gösterilir.
2. **Basitlik:** Karar ağaçları, genellikle diğer karmaşık modellere göre daha basittir. Bu, modelin eğitilmesi, anlaşılması ve uygulanması açısından avantaj sağlar.
3. **Özellik Seçimi:** Karar ağaçları, veri setindeki özelliklerin önem sıralamasını belirleme yeteneğine sahiptir. Bu, hangi özelliklerin modelin performansını artırdığını anlamak için faydalı olabilir.
4. **Kategorik ve Sayısal Değişkenlere Uyum:** Karar ağaçları, kategorik ve sayısal değişkenlere uyum sağlayabilir. Bu, karmaşık veri setlerini ele almak için idealdir.
5. **Veri Normalizasyonuna Duyarlılık:** Karar ağaçları, genellikle veri normalizasyonuna (örneğin, özelliklerin ölçeklendirilmesi) daha az duyarlıdır, bu da veri ön işleme sürecini basitleştirir.

**2.6. Dezavantajlar:**

1. **Aşırı Uyum (Overfitting):** Büyük ve karmaşık ağaçlar, eğitim veri setine aşırı uyum sağlama eğilimindedir. Bu durum, modelin yeni verilere uygulanabilirliğini azaltabilir.
2. **Hassasiyet:** Karar ağaçları, veri setindeki küçük değişikliklere karşı hassas olabilir. Bu durum, modelin istikrarsız olmasına ve değişken sonuçlar elde etmesine neden olabilir.
3. **Dengesiz Veri Setlerine Duyarlılık:** Dengesiz sınıflara sahip veri setlerinde, yani bir sınıfın diğerinden çok daha fazla örneğe sahip olduğu durumlarda performansları azalabilir.
4. **Çoklu Kararlar ile Karmaşıklık:** Büyük karar ağaçları, çok sayıda karar içerdiğinden ve genellikle çoklu düğümlere sahip olduğundan, modelin karmaşıklığını artırabilir ve anlaşılabilirliği azaltabilir.
5. **Tutarlılık Eksikliği:** Veri setinin küçük değişikliklere tepkisi öngörülemez olabilir, bu nedenle modelin tutarlı olma garantisi zordur.
6. **VERİ GİRİŞİ VE ÖN İŞLEME:**

Veri seti, Kaggle’dan bulup 5000 satırını aktardığım yerel Excel dosyasından alınmıştır. Veri seti, ev fiyatlarını etkileyen çeşitli özellikleri içerir. Bu özellikler arasında metrekare alanı, yatak odası sayısı, banyo sayısı, evin bulunduğu bölge/semt (kırsal-kentsel-kenar mahalle) ve yapım yılı bulunmaktadır. Veri seti içindeki kategorik veriler, model eğitimi için sayısal formata dönüştürülmüştür.

**Veri Seti (Örnek 5 Satır):**

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

Burada SquareFeet değişkenimizden başlayacak olursak, bu veri setinde bu değişkenin aldığı en büyük değer 2999 metrekare olurken en düşük değeri ise 1000 metrekare olmuştur. Yatak odası sayısı 2 ile 5 arasında bir değer alırken banyo sayısı da 1 ile 3 arasında bir değer alabilmektedir. Evin bulunduğu bölge değişkeni ise az önce de bahsettiğim gibi kırsal, kentsel ve kenar mahalle değişkenlerini almaktadır. Evin yapım yılı değişkeni ise 1950 ile 2021 arasında bir değer alabilir. Bu veri setinde en ucuz ev 36.6k değerlerinde iken en pahalı ev ise 492k para birimi değerindedir.

**Neighborhood (semt) kategorik verinin sayısal formata dönüştürülmesi:**

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

Burada görmüş olduğunuz gibi neighborhood değişkenini sayısal formata dönüştürdük. Bunu kodlarla da yapabilirdik ama ben excelden yapmayı tercih ettim. Buradaki 1 değeri rural yani kırsal, 2 değeri suburb yani kenar mahalle, 3 ise urban yani kentsel değerlerini taşıyor.

Bu 5 satırlık veri tablomuzda bir yorum yapacak olursak; ilk ve son satırdaki verileri ve evlerin değerlerini karşılaştıralım. Evlerin metrekare alanları aşağı yukarı aynı, yalnızca 4 birimlik bir fark var. Yatak odası ve banyo sayısında ise 1’er farkla alt satırdaki evimiz daha avantajlı. Yapım yılları olarak da yaklaşık 30 sene ile alt satırdaki evimiz daha yeni. Evlerin bulunduğu semtler ise ilk evimiz kırsalda iken, ikinci yani alt satırdaki evimiz kenar mahallede bulunmaktadır. Fiyat olarak ise ilk evimiz 215k değerinde iken ikinci evimiz 272k değerindedir. Yaklaşık 50k değerlik fiyat farkı bulunmaktadır.

1. **VERİ SETİ BÖLME VE MODEL EĞİTİMİ:**

Bu aşamada, veri setinin eğitim ve test setlerine ayrılması ve ardından karar ağacı modelinin eğitimi anlatılacaktır.

**4.1. Veri Seti Bölme:**

* Eğitim ve test setlerine bölme işlemi, projenin genel başarısını değerlendirmek için kritik bir adımdır. Veri setinin bu şekilde bölünmesi, modelin eğitim sırasında kullanılacak veri ile test edilmesi için ayrı bir veri setine sahip olmasını sağlar.
* Projede kullanılan **train\_test\_split** fonksiyonu, veriyi rastgele iki alt küme arasında böler. Bu, modelin genelleme yeteneğini test etmek için önemlidir.

**4.2. Karar Ağacı Modelinin Oluşturulması:**

Karar ağacı modeli, veri setindeki desenleri ve ilişkileri öğrenir. Bu model, eğitim seti üzerinde özellikleri ve hedef değişkeni içeren bir ağaç yapısı oluşturur.

**Bağımsız ve Bağımlı Değişkenlerin Belirlenmesi:**

* Veri setindeki bağımsız değişkenler (**x**) ve bağımlı değişken (**y**) doğru bir şekilde belirlenmiştir. Bu adım, modelin hangi özelliklere dayanarak fiyat tahminleri yapacağını belirler.

**4.3. Modelin Seçimi Eğitilmesi:**

* Model, eğitim seti üzerindeki özellikleri ve hedef değişkeni kullanarak öğrenir. Bu, karar ağacının dallarının ve düğümlerinin belirlenmesini sağlar.
* Projede karar ağacı regresyon modeli seçilmiş ve eğitilmiştir. Bu model, veri setindeki karmaşık ilişkileri öğrenmeye uygun ve anlaşılabilir bir modeldir.
* **DecisionTreeRegressor** sınıfı kullanılarak model oluşturulmuş ve **fit** fonksiyonu ile eğitim verilerine uyarlanmıştır.

Bu aşama, modelin temelini oluşturduğu için kritiktir. Veri setinin doğru şekilde bölünmesi ve modelin eğitimi, sonraki aşamalarda elde edilen sonuçların temelini oluşturur. Ayrıca, modelin eğitim performansının test seti üzerinde nasıl genelleme yapabileceğini anlamak, projenin başarısını belirlemede önemlidir.

**4.4. Test Verisi Üzerinde Tahmin Yapma:**

Eğitilen model, test seti üzerindeki özelliklere dayalı olarak hedef değişkeni tahmin eder. Bu tahminler, modelin gerçek dünya verilerinde nasıl performans göstereceğini belirlemede kullanılır.

**4.5. Model Performansının Değerlendirilmesi:**

Modelin performansı, tahminlerin gerçek değerlerle karşılaştırılması ile değerlendirilir. Bu, önceden ayrılmış olan test seti üzerinde gerçekleştirilir ve belirli bir performans metriği (örneğin, ortalama kare hata) kullanılarak ölçülür.

Bu adımların tamamı, modelin eğitiminden testine kadar olan süreci temsil eder. Bu noktada, modelin ne kadar iyi performans gösterdiğini anlamak için önceden belirlenmiş performans metrikleri kullanılır. Bu aşamada yapılan değerlendirmeler, modelin gerçek dünya verilerindeki başarısını tahmin etmede yardımcı olur.

1. **TAHMİN VE PERFORMANS DEĞERLENDİRMESİ:**

Bu aşamada, eğitilmiş karar ağacı modeli üzerinden tahminler yapılır ve modelin performansı değerlendirilir.

**5.1. Test Verisi Üzerinde Tahmin Yapma:**

Eğitilmiş karar ağacı modeli, test seti üzerindeki özelliklere dayanarak hedef değişkeni tahmin eder. Bu tahminler, modelin gerçek dünya verilerinde ne kadar iyi performans gösterdiğini belirtir.

**5.2. Performans Değerlendirmesi:**

Tahminlerin performansı, gerçek değerlerle karşılaştırılarak ölçülür. Performans metrikleri, modelin ne kadar doğru tahmin yaptığını değerlendirmek için kullanılır. Örneğin, ortalama kare hata (MSE) kullanılarak tahminler ile gerçek değerler arasındaki ortalama kare fark ölçülür. Düşük MSE değerleri, modelin daha iyi performans gösterdiğini belirtir.

**5.3. Yeni Veri ile Tahmin Yapma:**

Eğitilmiş model, öğrendiklerini kullanarak yeni, görülmemiş bir veri seti üzerinde de tahmin yapabilir. Bu, modelin gerçek dünya verilerini nasıl genelleştirebildiğini gösterir.

**5.4. Sonuçların Yorumlanması:**

Elde edilen sonuçlar, modelin performansını ve tahmin yeteneğini değerlendirir. Düşük hata değerleri ve gerçekleşen tahminlerin gerçek değerlere yakın olması, modelin güvenilir olduğunu gösterir. Modelin başarı kriterlerini karşılaması durumunda, başarıyla eğitildiği ve değerlendirildiği söylenebilir.

Bu aşamaların tamamlanmasıyla, projenin başlangıcında belirlenen amaçlara ulaşılmış ve modelin gerçek dünya verileri üzerinde başarılı tahminler yapma yeteneği değerlendirilmiştir.

1. **KARAR AĞACI GÖRSELLEŞTİRMESİ:**

Karar ağacı görselleştirmesi, modelin içsel yapısını anlamak ve karar alma sürecini görsel olarak temsil etmek amacıyla kullanılır. Şu adımları içerir:

**6.1. Karar Ağacını Oluşturma:**

* Eğitilmiş karar ağacı modeli içsel bir ağaç yapısı oluşturur.

**6.2. Görselleştirme Aracını Kullanma:**

* Görselleştirmek için çeşitli araçlar bulunmaktadır, bunlar genellikle karar ağacı modellerini anlamak ve görselleştirmek için kullanılan araçlardır. Graphviz gibi araçlar önerilir.

**6.3. Görselleştirmenin Anlamını Çıkartma:**

* Görselleştirmede her düğüm, bir özelliği ve bu özellikten yapılan bir kararı temsil eder. Dallar ve yaprak düğümler, karar ağacının belirli özellik değerlerine göre nasıl bölündüğünü ve tahminlerin nasıl yapıldığını gösterir.

**6.4. Görselleştirmenin Yorumlanması:**

* Görselleştirme, modelin içsel karar süreçlerini anlamak için kullanılır. En üst düğümden başlayarak her düğüm ve dal takip edilerek modelin kararlarını yorumlamak mümkündür.

**6.5. İş Akışı ve Genel Anlam:**

* Görselleştirmenin temel amacı, modelin karmaşık içsel yapısını anlamayı kolaylaştırmak ve modelin nasıl kararlar verdiğini açıklamak. Bu, modelin kullanıcılar veya paydaşlar tarafından daha iyi anlaşılmasını sağlar.

Bu şekilde, karar ağacı görselleştirmesinin önemini, kullanımını ve elde edilen görselin nasıl yorumlanacağını anlamak mümkündür. Görselleştirmeler, modelin açıklanabilirliğini artırarak güvenilirliğini ve anlaşılabilirliğini artırır.

Bu projede elde edilen karar ağacı görselleştirmesine dayanarak şu yorumları yapabilirim:

* **Karmaşıklık ve Basitleme:** Karar ağacı modelinin genel yapısı oldukça anlaşılır ve basit görünmektedir. Bu, ev fiyatlarını tahmin etmek için temel özelliklerin (metrekare, yatak odası sayısı vb.) nasıl kullanıldığını anlamayı kolaylaştırır.
* **Önemli Özelliklerin Belirlenmesi:** Görselleştirmeden anlaşıldığı kadarıyla, karar ağacı belirli özelliklerin (örneğin, SquareFeet) ev fiyat tahmininde büyük bir etkiye sahip olduğunu vurgulamaktadır. Bu, kullanıcılar ve paydaşlar için değerli bir bilgi sağlar.
* **Veri Bölünmeleri ve Kararlar:** Görselleştirme, veri setindeki bölünmeleri ve her düğümde alınan kararları gösterir. Özellikle, karar ağacının belirli değerlere göre nasıl dallandığını anlamak, modelin hangi koşullarda hangi tahminleri yaptığını gösterir.
* **Açıklanabilirlik ve İnceleme:** Karar ağacı, modelin tahminlerini anlamak için kullanıcılar ve diğer paydaşlar için açıklayıcı bir araçtır. Her bir düğümün anlamını kavramak, modelin nasıl çalıştığını daha iyi anlamamıza yardımcı olur.
* **Aykırı Değerlere Karşı Hassasiyet:** Görselleştirme, modelin aykırı değerlere karşı nasıl tepki verdiğini göstermez, ancak bu durum karar ağaçları için genellikle bir zayıflık olabilir. Modelin bu tür durumlarla başa çıkma yeteneği dikkate alınmalıdır.
* **Genelleme Yeteneği:** Karar ağacının genelleme yeteneği, veri setine ve projenin bağlamına bağlı olarak değerlendirilmelidir. Görselleştirme, modelin ne kadar genel ve uygulanabilir olduğunu anlamada bir gösterge sağlar.

Bu yorumlar, karar ağacı görselleştirmesinin sağladığı değeri vurgular ve modelin içsel yapısını anlamak için bu tür araçların kullanımının önemini belirtir.

1. **SONUÇ VE YORUM:**

Bu projede, bir karar ağacı regresyon modeli kullanılarak ev fiyatlarının tahmin edilmesi amaçlandı. Proje adımları şu şekilde gerçekleşti:

**7.1. Performansın Değerlendirilmesi:**

Modelin performansını değerlendirmek için ortalama kare hata (MSE) kullanıldı. Elde edilen düşük MSE değerleri, modelin eğitim ve test setlerinde başarılı tahminler yaptığını gösteriyor. Ancak, modelin hiperparametre ayarları ve daha fazla veriyle iyileştirilebilecek potansiyel bir alan olduğunu belirtmek önemlidir.

**7.2. Modelin Güçlü Yönleri:**

Model, özellikle belirli özelliklerin (örneğin, metrekare, yatak odası sayısı) fiyat tahmininde önemli bir etkiye sahip olduğunu başarıyla öğrenmiştir. Ayrıca, karmaşık veri ilişkilerini öğrenme yeteneği, modelin güçlü yönlerinden biridir.

**7.3. Modelin Zayıf Yönleri:**

Model, bazı durumlarda daha yüksek hatalar yapabilir ve özellikle aykırı değerlere karşı hassas olabilir. Ayrıca, veri setinde eksik veya yanıltıcı veriler olduğunda performansı etkilenebilir.

**7.4. İyileştirme Önerileri:**

Modelin performansını artırmak için, hiperparametre ayarları daha dikkatli bir şekilde yapılabilir. Ayrıca, daha fazla veri toplama veya özellik mühendisliği ile modelin genelleme yeteneği artırılabilir.

**7.5. Uygulanabilirlik ve Genelleme:**

Modelin uygulanabilirliği, ev fiyatlarını tahmin etmek için başlangıçta belirlenen amaca hizmet etmektedir. Ancak, modelin farklı coğrafi bölgelerde veya farklı pazar koşullarında genelleme yeteneği dikkate alınmalıdır.

**7.6. Sonuçların Projedeki Yerine Göre Yorumlanması:**

Projenin başlangıcındaki hedeflere ulaşıldığı söylenebilir. Model, ev fiyatları üzerinde etkili tahminler yapabiliyor ve bu, projenin temel amacına hizmet ediyor.

**7.7. Projenin Genel Değerlendirmesi:**

Proje, ev fiyatları tahmininde kullanılan karar ağacı regresyon modeli ile başarılı bir şekilde sonuçlandı. Modelin kullanılabilirliği ve anlaşılabilirliği, projenin genel başarısına katkı sağladı.

**7.8. Sonuçlar Üzerinden Strateji Belirleme:**

Elde edilen sonuçlar doğrultusunda, modelin performansını daha da artırmak ve genelleme yeteneğini güçlendirmek için iyileştirmeler yapılabilir. Ayrıca, benzer projelerde kullanılabilirlik ve genelleme potansiyelini artırmak için stratejiler belirlenebilir.

Bu projenin sonuçları, ev fiyatlarını tahmin etme konusunda kullanılan karar ağacı modelinin etkili bir araç olduğunu gösteriyor. Ancak, her model gibi, bu modelin de belirli sınırlamaları ve iyileştirme potansiyelleri bulunmaktadır.

**KODLAR:**

from google.colab import files

uploaded = files.upload()

import pandas as pd

import io

# Yüklenen dosyayı oku

data = pd.read\_excel(io.BytesIO(uploaded['ev fiyat 5000.xlsx']))

# Veriyi kontrol etmek için ilk birkaç satırı yazdırabilirsiniz

print(data.head())

satir\_sayisi, sutun\_sayisi = data.shape

print(f"Satır Sayısı: {satir\_sayisi}, Sütun Sayısı: {sutun\_sayisi}")

df = pd.DataFrame(data)

SquareFeet Bedrooms Bathrooms Neighborhood YearBuilt Price

0 2126 4 1 1 1969 215355

1 2459 3 2 1 1980 195014

2 1860 2 1 2 1970 306891

3 2294 2 1 3 1996 206786

4 2130 5 2 2 2001 272436

Satır Sayısı: 5000, Sütun Sayısı: 6

# Hedef değişken ve bağımsız değişkenleri belirleme

X = df[['SquareFeet', 'Bedrooms', 'Bathrooms', 'Neighborhood', 'YearBuilt']]

y = df['Price']

# Veriyi eğitim ve test setlerine bölelim

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2)

# Karar ağacı modelini oluşturma

from sklearn.tree import DecisionTreeRegressor

model = DecisionTreeRegressor(random\_state=42)

model.fit(X\_train, y\_train)

# Test seti üzerinde tahmin yapma

y\_pred = model.predict(X\_test)

# Model performansını değerlendirme

print(model.score(X\_train,y\_train))

print(model.score(X\_test,y\_test))

from sklearn.metrics import r2\_score

r2 = r2\_score(y\_test, y\_pred)

print(f"R^2 Score: {r2}")

from sklearn.metrics import mean\_squared\_error

mse = mean\_squared\_error(y\_test, y\_pred)

print(f"Ortalama Kare Hata (MSE): {mse}")

0.9999444842086512

0.05822837697478356

R^2 Score: 0.05822837697478356

Ortalama Kare Hata (MSE): 5545597664.426

# Yeni bir ev için fiyat tahmini için verileri girdik

new\_data = {

    'SquareFeet': [1161], # 1000-2999

    'Bedrooms': [4], # 2-5

    'Bathrooms': [2], # 1-3

    'Neighborhood': [2],

    'YearBuilt': [2003],

}

new\_df = pd.DataFrame(new\_data)

# Tahmin yapın

predicted\_price = model.predict(new\_df)

print(f"Yeni ev için tahmini fiyat: {predicted\_price[0]}")

Yeni ev için tahmini fiyat: 288571.0

# karar ağacı görselleştirme

!pip install pydot

!apt-get install graphviz

# Ardından aşağıdaki kodu kullanarak karar ağacını görselleştirdik

from sklearn.tree import export\_graphviz

import graphviz

from IPython.display import display

#Regresyon İçin Karar Ağaçları

from sklearn.tree import  DecisionTreeRegressor

model\_reg =  DecisionTreeRegressor(max\_depth=2)

model\_reg.fit(X\_train,y\_train)

export\_graphviz(model\_reg, out\_file="mytree1.dot",filled=True)

with open("mytree1.dot") as f:

    dot\_graph = f.read()

graphviz.Source(dot\_graph)

metin, yazı tipi, çizgi, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu