**TAKIM ADI:** SoftTech

**TAKIM ÜYELERİ:**

1. Hakan Demirli
2. Ahmet Onay
3. Murat Yurdugül
4. Berşan Akgül

## 1. PROJE ÖZETİ

Projemiz, 2020 yılı İstanbul konut piyasası verilerini kullanarak, özellikle sabit gelirli ve krediye uygun konut arayan alıcılar için geliştirilmiş hibrit bir değerleme sistemidir. Yalnızca standart bir regresyon modeli kullanmak yerine, modelin hata paternlerini öğrenen "OOF (Out-of-Fold) Correction" mekanizması ve m² tabanlı segmentasyon ile güçlendirilmiş, %20 gizli test setindeki sapmaları minimize eden akıllı bir yatırım danışmanıdır.

## 2. KURULUM VE ÇALIŞTIRMA

**Projeyi Çalıştırma Adımları**

**Ön Gereksinimler**

* **Python 3.x** yüklü olmalı (Python 3.8 veya üstü önerilir)

**Kurulum Adımları**

bash

# 1. Proje klasörüne git

cd Soft\_tech\_ml

# 2. Gerekli paketleri yükle

pip install -r requirements.txt

# 3. Uygulamayı başlat

python app.py

**Kullanım**

Uygulama başladıktan sonra tarayıcıda şu adresi aç:

plaintext

http://localhost:5000

veya

plaintext

http://127.0.0.1:5000

**requirements.txt İçeriği**

Eğer requirements.txt dosyası yoksa veya eksikse, şu paketler gerekli:

txt

flask

flask-cors

pandas

scikit-learn

xgboost

numpy

joblib

scipy

**Önemli Notlar**

1. **Model dosyaları** models/ klasöründe olmalı:
   * investment\_advisor\_model\_v7.pkl
   * location\_data\_v7.pkl
2. **Port 5000** kullanılıyor, eğer başka bir uygulama bu portu kullanıyorsa çakışma olabilir
3. **İlk çalıştırmada** model yüklenirken biraz zaman alabilir (1-2 saniye)
4. **Terminal çıktısı** şöyle görünmeli:

plaintext

Model yüklendi!

Model Performance:

Test MAPE: 18.26%

Test R²: 0.8201

Application initialized successfully!

Starting server on http://0.0.0.0:5000

**🛠️ Sorun Giderme**

**Problem**: "Module not found" hatası  
**Çözüm**: pip install -r requirements.txt komutunu tekrar çalıştır

**Problem**: "Port already in use" hatası  
**Çözüm**: Port 5000'i kullanan uygulamayı kapat veya app.py içinde farklı port kullan

**Problem**: Model dosyası bulunamadı  
**Çözüm**: models/ klasörünün ve içindeki .pkl dosyalarının varlığını kontrol et

## 3. VERİ ÖN İŞLEME YAKLAŞIMI

Veri setindeki gürültüyü azaltmak ve modelin genelleme yeteneğini artırmak için kapsamlı bir temizlik ve özellik mühendisliği süreci uyguladık.

* **Temizlenen Alanlar:**
  + **Price:** "TL", nokta ve virgül gibi karakterler temizlendi, sayısal formata çevrildi.
  + **Kredi Uygunluğu Filtresi:** Hedef kitle analizi gereği sadece 'Available for Loan' == 'Yes' olan kayıtlar filtrelendi.
  + **Outlier Temizliği:** Fiyat dağılımında %1 ve %99'luk dilimlerin dışında kalan uç değerler veri setinden çıkarıldı.
  + **Metin Ayrıştırma:** 'Building Age' (ör. "5-10 between"), 'Floor location' (ör. "High entrance") ve 'Number of rooms' (ör. "3+1") sütunları sayısal verilere dönüştürüldü.
* **Eklenen Yeni Özellikler (Feature Engineering):**
  + **Skorlama Sistemi:** Evin özelliklerine göre 'Luxury\_Score' (Havuz, Sauna vb.), 'Comfort\_Score' (Asansör, Balkon vb.) ve 'Security\_Score' türetildi.
  + **Tarihsel Özellikler:** İlan tarihi parse edilerek 'Days\_Since\_Ref' (Referans tarihe uzaklık), 'Ad\_Month' (Mevsimsellik) özellikleri eklendi.
  + **Target Encoding:** 'District' ve 'Neighborhood' gibi yüksek kardinaliteli kategorik veriler, K-Fold Target Encoding yöntemiyle (Smoothing uygulanarak) sayısal hale getirildi.
  + **m² Segmentasyonu:** Fiyat sızıntısını (leakage) önlemek için fiyata dayalı değil, evin büyüklüğüne dayalı (m² Net) 5 farklı segment oluşturuldu.
* **Seçilen Kritik Özellikler (Feature Selection):**
  + Interaction Features: Özelliklerin birbirleriyle etkileşimi modellendi (ör. 'm2\_x\_District', 'Luxury\_x\_District', 'Age\_Decay'). Toplamda model performansına en çok katkı sağlayan etkileşimler seçildi.

## 4. MODEL MİMARİSİ

Standart modellemenin ötesine geçerek, modelin sistematik hatalarını düzelten iki aşamalı bir mimari kurguladık.

* **Kullanılan Algoritma:** XGBoost Regressor + OOF (Out-of-Fold) Correction Layer.
* **Neden bu algoritma?:** XGBoost, tabular verilerde ve doğrusal olmayan ilişkilerde en yüksek performansı vermektedir. Ancak tek başına yeterli görmedik; modelin eğitim setinde göremediği verilerdeki sapmasını (bias) tespit etmek için "OOF Correction" katmanı ekledik. Bu katman, modelin belirli segmentlerde (örneğin lüks semtlerdeki küçük evlerde) yaptığı sistematik hataları öğrenip son tahmini düzeltmektedir.
* **Elde Edilen Başarı Skoru:**
  + Model, eğitim sırasında ağırlıklandırılmış örneklem (Sample Weighting) kullanarak az bulunan ev tiplerini de öğrenmiştir.
  + Hiyerarşik düzeltme sonrası MAPE (Ortalama Mutlak Yüzde Hata) ve R-Kare skorlarında, baz modele göre belirgin iyileşme sağlanmıştır.

## 5. YATIRIM KARAR MANTIĞI

Sistemimiz sadece tek bir fiyat tahmini yapmaz; her ev için dinamik bir "Güven Aralığı" (Confidence Interval) hesaplar. Bu aralık, o segmentteki modelin tarihsel hata payına göre belirlenir.

* **Adil Değer (Fair Value):** Modelin ham tahmini + Correction (Düzeltme) Payı.
* **Güven Aralığı Hesabı:** Fair Value üzerinden, o evin bulunduğu segmentin hata varyansına göre Alt ve Üst sınırlar belirlenir.

Karar mekanizması şu şekildedir:

1. **FIRSAT Eşiği (Mükemmel/İyi Fırsat):** Eğer İlan Fiyatı < Alt Sınır ise; sistem evin değerinin altında satıldığını tespit eder ve "FIRSAT" uyarısı verir. Güven aralığı darsa "Mükemmel Fırsat", genişse "İyi Fırsat" olarak etiketlenir.
2. **PAHALI Eşiği (Çok Pahalı/Pahalı Görünüyor):** Eğer İlan Fiyatı > Üst Sınır ise; ev piyasa değerinin üzerindedir ve "PAHALI" olarak etiketlenir.
3. **ADİL FİYAT (Değerlendir/Pazarlık Yap):** İlan fiyatı Alt ve Üst sınırlar arasındaysa, fiyat piyasa normallerindedir. Sistem "Adil Fiyat" sonucunu döner.

## 6. SİZİ DİĞERLERİNDEN AYIRAN ÖZELLİK

Projemizi rakiplerden ayıran en temel özellik, geliştirdiğimiz **"Hiyerarşik Post-Processing Correction"** sistemidir:

1. **Data Leakage Önlemi:** Çoğu model "Fiyat Segmenti"ni eğitimde kullanıp hataya düşerken (geleceği görmek), biz sızıntısız "m² Segmenti" kullandık.
2. **Akıllı Düzeltme Tablosu:** Modelimiz tahmin yaptıktan sonra, evin özelliklerine (Lüks durumu, m² grubu, Semt tipi) bakarak geçmişte bu tip evlerde ne kadar hata yaptığını "Correction Table"dan sorgular ve tahmini revize eder.
3. **Dinamik Mahalle Düzeltmesi:** Eğer bir mahallede yeterli veri varsa o mahalleye özel düzeltme katsayısı, yoksa ilçe geneli, o da yoksa segment geneli kullanılarak "Hiyerarşik" bir yaklaşım sergilenir. Bu sayede veri azlığında bile tutarlı tahminler üretilir.