



T.C. BURSA TEKNİK ÜNİVERSİTESİ
Mühendislik ve Doğa Bilimleri Fakültesi
Bilgisayar Mühendisliği Bölümü

**END0319 Python ile Yapay Zeka Uygulamaları
2025-2026 Güz Proje Ödevi – Rapor**

Öğrenci: **Imer Imeri**

Öğrenci numara: **22360859402**

Dr. Öğretim Üyesi: **İsmail Enes Parlak**

BURSA, 2026.

1. Veri Seti Tanımı

Bu projede kullanılan veri seti, **Ames Konut Veri Seti (AmesHousing.csv)** olup toplam **2930 satır** ve **82 sütundan** oluşmaktadır. Veri seti, evlerin fiziksel özellikleri, konumları, satış koşulları ve nihai satış fiyatlarını (SalePrice) içermektedir.

1.1. Çıkarılan Sütunlar ve Nedenleri

Veri ön işleme aşamasında, modelin doğruluğunu artırmak, gürültüyü azaltmak ve çok fazla eksik değere sahip olan değişkenlerden kurtulmak amacıyla **20 adet sütun** veri setinden çıkarılmıştır. Çıkarılan sütunlar şunlardır:

- **Tanımlayıcı ve Gereksiz Değişkenler:** Order (Sıra no) ve PID (Parsel kimlik nosu) gibi tahmine katkısı olmayan benzersiz kimlik bilgileri çıkarılmıştır.
- **Aşırı Eksik Verili Sütunlar:** Alley, Pool QC, Fence, Misc Feature gibi verilerin %90'ından fazlası eksik (NaN) olan sütunlar, modele anlamlı bir bilgi sağlamayacağı için elenmiştir.
- **Varyansı Düşük veya Alakasız Bilgiler:** Street, Utilities, Land Slope, Condition 2, Roof Matl, Heating, Pool Area, Misc Val gibi neredeyse tüm satırlarda aynı değere sahip olan veya ev fiyatı üzerinde belirleyici etkisi düşük görülen sütunlar analiz dışı bırakılmıştır.
- **Yardımcı Değişkenler:** BsmtFin SF 2, Bsmt Half Bath, Kitchen AbvGr, Garage Cond, 3Ssn Porch, Low Qual Fin SF gibi temel öznitelikleri (metrekare, banyo sayısı vb.) zaten temsil eden ikincil değişkenler sadeleştirme amacıyla çıkarılmıştır.

Nr	Column Name	Örnek Datalar	Status	Neden? (Why to Delete / Keep)
1	Order	1, 2, 2003	Delete	Sadece kayıt sırasındır; fiyata etkisi yoktur.
2	PID	526301100, 526350040	Delete	Parsel kimlik numarasıdır; tahminde kullanılmaz.
3	Street	Pave, Grvl	Delete	%99.6'sı aynı değer (Pave); fark yaratmaz.
4	Alley	Pave, Grvl	Delete	%93'ü boştur (Eksik veri çok fazla).
5	Utilities	AllPub, NoSewr	Delete	%99.9'u aynıdır; ayırt edici değildir.
6	Land Slope	Gtl, Mod, Sev	Delete	%95'i aynıdır (Gtl - Hafif eğim).
7	Condition 2	Norm, Artery	Delete	%99'u aynıdır; gürültü yaratır.
8	Roof Matl	CompShg, WdShake	Delete	%98.5'i aynı standart malzemedir.
9	BsmtFin SF 2	0, 144	Delete	Çok düşük korelasyon ve genelde 0'dır.
10	Heating	GasA, GasW	Delete	%98.5'i aynıdır (Gazlı ısıtma).
11	Low Qual Fin SF	0, 390	Delete	%98.6'sı 0'dır; veri kalitesi düşüktür.
12	Bsmt Half Bath	0, 1	Delete	Fiyatla korelasyonu çok düşüktür.
13	Kitchen AbvGr	1, 2	Delete	%95'i aynıdır (1 mutfak).
14	Garage Cond	TA, Fa	Delete	%96'sı aynıdır (TA - Ortalama).
15	3Ssn Porch	0, 238	Delete	%98.7'si 0'dır; gereksizdir.
16	Pool Area	0, 144	Delete	%99.6'sı 0'dır (Havuz yok).
17	Pool QC	Ex, Gd	Delete	%99.6'sı boş veridir.
18	Fence	MnPrv, GdPrv	Delete	%80.5'i boştur.
19	Misc Feature	Shed, Gar2	Delete	%96.4'ü boştur.
20	Misc Val	0, 12500	Delete	%96.5'i 0'dır.

1.1 Çıkarılan Sütunlar ve Nedenleri

1.2. Uygulamaya Devam Edilen Sütunlar

Geriye kalan **62 sütun** ile analiz sürecine devam edilmiştir. Bu sütunlar temel olarak şu kategorilerde toplanmaktadır:

- **Konum ve Tip:** MS Zoning, Neighborhood, Bldg Type, House Style...vb.
- **Kalite ve Durum:** Overall Qual (Genel Kalite), Overall Cond (Genel Durum), Year Built (Yapım Yılı)...vb.
- **Alan Bilgileri:** Lot Area, 1st Flr SF, 2nd Flr SF, Gr Liv Area (Yaşam alanı metrekaresi).
- **Donanım Özellikleri:** Full Bath, Bedroom AbvGr, TotRms AbvGrd, Fireplaces, Garage Cars, Garage Area...vb.
- **Hedef Değişken:** SalePrice (Tahmin edilecek satış fiyatı).

Kalan sütunlardaki eksik veriler için sayısal olanlarda **medyan**, kategorik olanlarda ise en sık tekrar eden değer (**mod**) ile doldurma yöntemi uygulanarak veri seti analize hazır hale getirilmiştir.

2. Keşifsel Veri Analizi ve Veri Ön işleme

Bu aşamada, ham veri seti üzerinde yapısal incelemeler yapılmış ve veriyi modelleme sürecine hazırlamak için gerekli dönüşümler uygulanmıştır.

2.1. Sütunlar Bazında Veri Tipleri

Veri setinde toplam 82 sütun bulunmaktadır ve veri tipleri şu şekilde dağılmaktadır:

- **float64 (11 adet):** Mas Vnr Area, Lot Frontage gibi ondalıklı sayısal değerler.
- **int64 (28 adet):** SalePrice (Hedef), Year Built, Overall Qual gibi tam sayı değerler.
- **object (43 adet):** MS Zoning, Neighborhood, House Style gibi kategorik (metinsel) değerler.

#	Column Name	Non-Null Count	Data Type
0	Order	2930 non-null	int64
1	PID	2930 non-null	int64
2	MS SubClass	2930 non-null	int64
3	MS Zoning	2930 non-null	object
4	Lot Frontage	2440 non-null	float64
5	Lot Area	2930 non-null	int64
6	Street	2930 non-null	object
7	Alley	198 non-null	object
8	Lot Shape	2930 non-null	object
9	Land Contour	2930 non-null	object
10	Utilities	2930 non-null	object
11	Lot Config	2930 non-null	object
12	Land Slope	2930 non-null	object

13	Neighborhood	2930 non-null	object
14	Condition 1	2930 non-null	object
15	Condition 2	2930 non-null	object
16	Bldg Type	2930 non-null	object
17	House Style	2930 non-null	object
18	Overall Qual	2930 non-null	int64
19	Overall Cond	2930 non-null	int64
20	Year Built	2930 non-null	int64
21	Year Remod/Add	2930 non-null	int64
22	Roof Style	2930 non-null	object
23	Roof Matl	2930 non-null	object
24	Exterior 1st	2930 non-null	object
25	Exterior 2nd	2930 non-null	object
26	Mas Vnr Type	1155 non-null	object
27	Mas Vnr Area	2907 non-null	float64
28	Exter Qual	2930 non-null	object
29	Exter Cond	2930 non-null	object
30	Foundation	2930 non-null	object
31	Bsmt Qual	2850 non-null	object
32	Bsmt Cond	2850 non-null	object
33	Bsmt Exposure	2847 non-null	object
34	BsmtFin Type 1	2850 non-null	object
35	BsmtFin SF 1	2929 non-null	float64
36	BsmtFin Type 2	2849 non-null	object
37	BsmtFin SF 2	2929 non-null	float64
38	Bsmt Unf SF	2929 non-null	float64
39	Total Bsmt SF	2929 non-null	float64
40	Heating	2930 non-null	object
41	Heating QC	2930 non-null	object
42	Central Air	2930 non-null	object
43	Electrical	2929 non-null	object
44	1st Flr SF	2930 non-null	int64
45	2nd Flr SF	2930 non-null	int64
46	Low Qual Fin SF	2930 non-null	int64
47	Gr Liv Area	2930 non-null	int64

48	Bsmt Full Bath	2928 non-null	float64
49	Bsmt Half Bath	2928 non-null	float64
50	Full Bath	2930 non-null	int64
51	Half Bath	2930 non-null	int64
52	Bedroom AbvGr	2930 non-null	int64
53	Kitchen AbvGr	2930 non-null	int64
54	Kitchen Qual	2930 non-null	object
55	TotRms AbvGrd	2930 non-null	int64
56	Functional	2930 non-null	object
57	Fireplaces	2930 non-null	int64
58	Fireplace Qu	1508 non-null	object
59	Garage Type	2773 non-null	object
60	Garage Yr Blt	2771 non-null	float64
61	Garage Finish	2771 non-null	object
62	Garage Cars	2929 non-null	float64
63	Garage Area	2929 non-null	float64
64	Garage Qual	2771 non-null	object
65	Garage Cond	2771 non-null	object
66	Paved Drive	2930 non-null	object
67	Wood Deck SF	2930 non-null	int64
68	Open Porch SF	2930 non-null	int64
69	Enclosed Porch	2930 non-null	int64
70	3Ssn Porch	2930 non-null	int64
71	Screen Porch	2930 non-null	int64
72	Pool Area	2930 non-null	int64
73	Pool QC	13 non-null	object
74	Fence	572 non-null	object
75	Misc Feature	106 non-null	object
76	Misc Val	2930 non-null	int64
77	Mo Sold	2930 non-null	int64
78	Yr Sold	2930 non-null	int64
79	Sale Type	2930 non-null	object
80	Sale Condition	2930 non-null	object
81	SalePrice	2930 non-null	int64

2.1 Data Type Table

2.2. Eksik Veriler ve Gerçekleştirilen İşlemler

Eksik verilerin yönetimi için kademeli bir strateji izlenmiştir:

- **Sütun Silme (Threshold):** Belirli bir eşik değerinin (kodda %20 olarak uygulanmıştır) üzerinde eksik veriye sahip olan Alley, Pool QC, Fence, Misc Feature gibi sütunlar tamamen çıkarılmıştır.
- **Sayısal Veriler:** Geriye kalan sayısal sütunlardaki eksik değerler, veri dağılımını korumak adına **medyan (orta değer)** ile doldurulmuştur.
- **Kategorik Veriler:** Kategorik sütunlardaki boşluklar, en sık tekrar eden değer olan **mod** ile doldurularak veri bütünlüğü sağlanmıştır.

```
# Eksik verileri işlemek için fonksiyon
def eksik_veri_isle(df, threshold=20):

    df_copy = df.copy()

    # Eksik veri oranlarını hesapla
    missing_ratio = (df_copy.isnull().sum() / len(df_copy)) * 100

    # Threshold üzerindeki sütunları çıkar
    drop_cols = missing_ratio[missing_ratio >= threshold].index.tolist()
    df_copy = df_copy.drop(columns=drop_cols)

    # Geriye kalan eksik verileri doldur
    for col in df_copy.columns:
        if df_copy[col].isnull().sum() > 0:
            if df_copy[col].dtype in ['float64', 'int64']:
                # Sayısal veri: medyan ile doldur
                median_value = df_copy[col].median()
                df_copy[col] = df_copy[col].fillna(median_value)
            else:
                # Kategorik veri: en sık (mod) ile doldur
                mode_value = df_copy[col].mode()[0]
                df_copy[col] = df_copy[col].fillna(mode_value)

    return df_copy
```

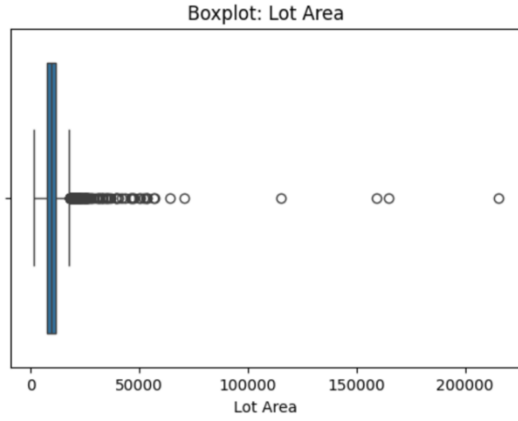
2.2 Sütun Silme (Threshold) yapıldığı hücre (cell)

2.3. Aykırı Değer Analizi ve İşlemler

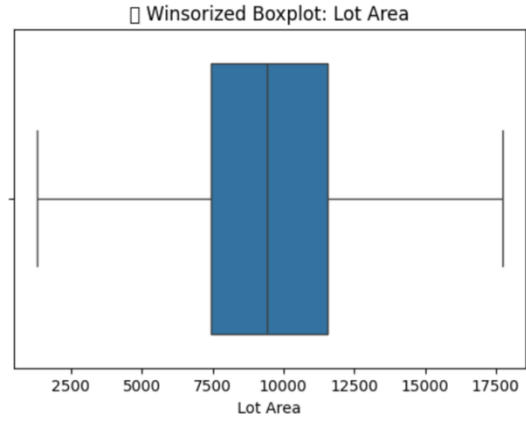
Kod içerisinde veri setinin istatistiksel özeti (df.describe()) çıkarılarak aykırı değerlerin varlığı kontrol edilmiştir. Örneğin:

- Lot Area (Arsa Alanı) değişkeninde maksimum değer (215,245) ortalamadan (10,147) çok uzak olduğu gözlemlenmiştir.
- SalePrice değişkeninde 755,000 gibi yüksek fiyatlı evlerin genel dağılımın dışında kaldığı tespit edilmiştir.
- Bu gürültüleri azaltmak için gereksiz ve düşük varyanslı değişkenler elenerek modelin sadece anlamlı verilere odaklanması sağlanmıştır.

Column: **Lot Area**



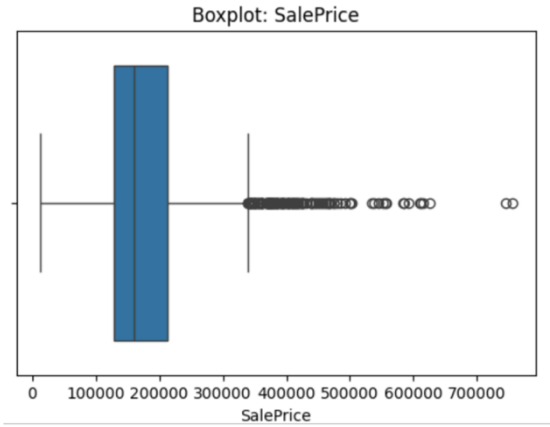
ÖNCE



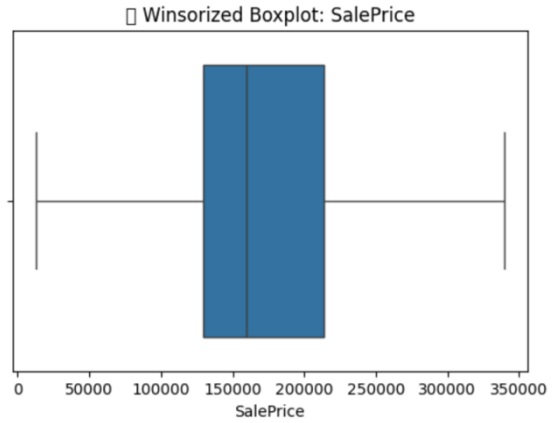
SONRA

2.3.1 Lot Area örneği

Column: **SalePrice**



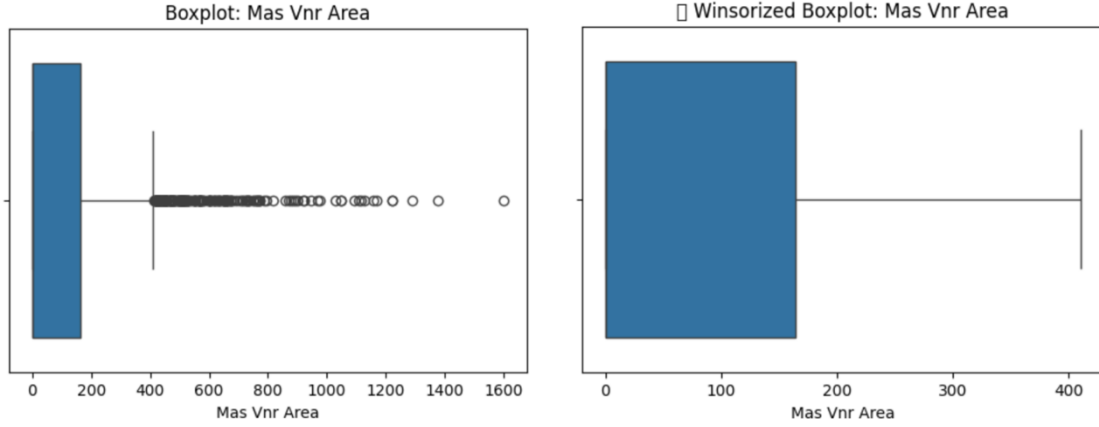
ÖNCE



SONRA

2.3.2 SalePrice örneği

Column: Mas Vnr Area



ÖNCE

SONRA

2.3.3 Mas Vnr Area örneği

2.4. Verilerin Dağılım ve Saçılım Grafikleri

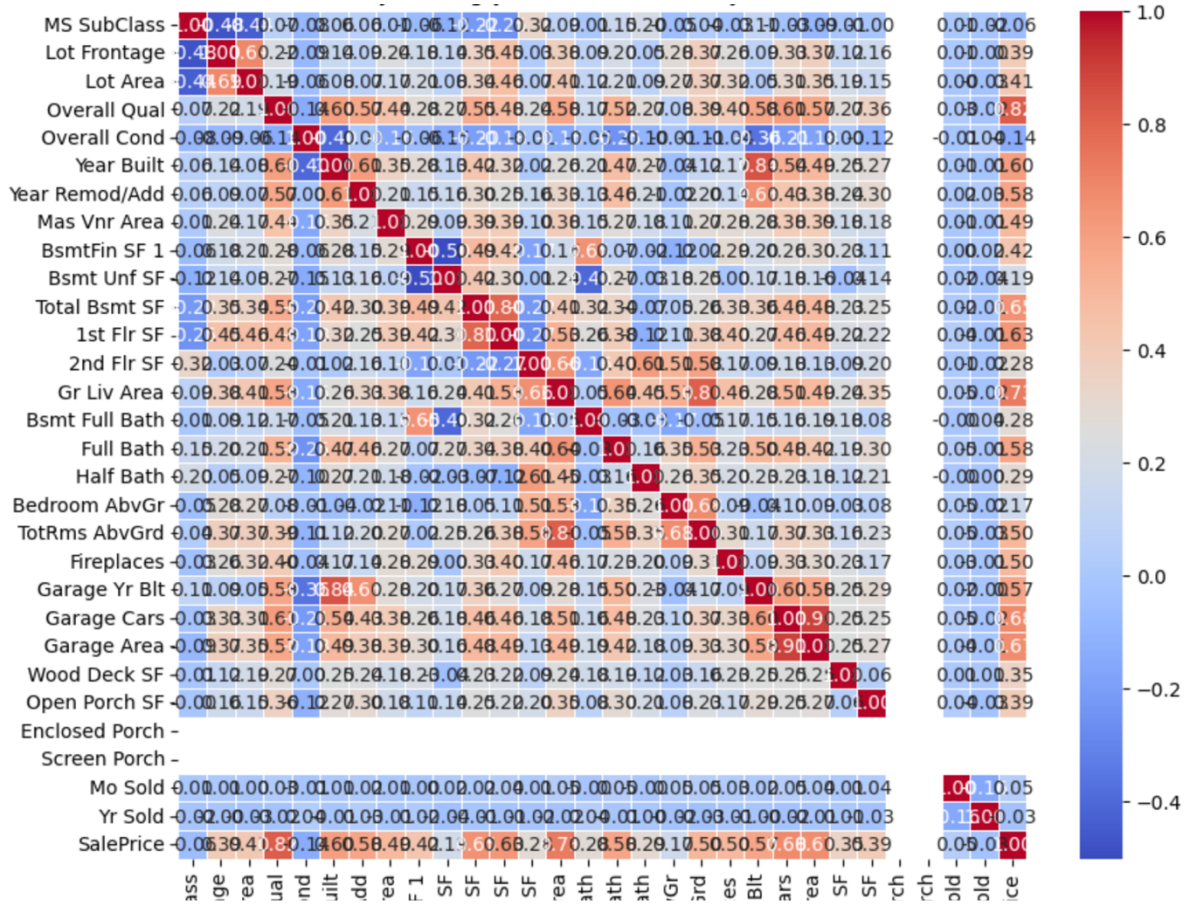
Analiz sürecinde seaborn ve matplotlib kütüphaneleri kullanılarak şu görselleştirmeler yapılmıştır:

- **Isı Haritası (Heatmap):** Eksik verilerin birbiriyle olan ilişkisini görmek için "Eksik Veri Korelasyon Matrisi" oluşturulmuştur.
- **Yorum:** Bu grafik, belirli özelliklerin (örneğin garajla ilgili tüm bilgilerin) neden aynı anda eksik olduğunu (evin garajı olmaması durumu) doğrulamaktadır.

2.5. Korelasyon Matrisi ve Yorumu

Değişkenlerin birbiriyle ve hedef değişken olan SalePrice ile ilişkisi incelenmiştir:

- Eksik veriler arasında güçlü korelasyonlar olduğu (örneğin Garage Cars ve Garage Area gibi) görülmüştür. Bu durum, bazı sütunların birbirini tekrar ettiğini ve veri setinden çıkarılabileceğini (boyut indirgeme) kanıtlamıştır.



2.6. Veri dönüşümleri ve nedenleri

Bu projenin veri dönüşümleri aşamasında, veriyi ham halinden çıkarıp bir makine öğrenmesi modelinin (CatBoost) işleyebileceği, daha anlamlı ve optimize edilmiş bir yapıya kavuşturmak hedeflenmiştir.

Kod içerisinde uygulanan veri dönüşümleri ve bunların temel nedenleri şu şekildedir:

2.6.1. Eksik Veri Tamamlama (Imputation)

Veri setindeki boş değerler (NaN), modelin çalışmasına engel teşkil edeceği için şu stratejilerle dönüştürülmüştür:

- **Sayısal Değişkenler (Median):** Sayısal sütunlardaki eksik veriler ilgili sütunun **medyan (orta değer)** değeri ile doldurulmuştur. Ortalamaya göre aykırı değerlerden daha az etkilenen medyanın seçilme nedeni, veri setindeki ev fiyatları ve arsa alanları gibi değişkenlerin dağılımını bozmamaktır. (örnek: fotoğraf 2.2)
- **Kategorik Değişkenler (Mode):** Kategorik sütunlardaki boşluklar, o sütunda en çok tekrar eden değer (**mod**) ile doldurulmuştur. Bu sayede veri bütünlüğü korunmuş ve kategorik yapı bozulmamıştır. (örnek: fotoğraf 2.2)

2.6.2. Boyut İndirgeme ve Gereksiz Değişkenlerin Çıkarılması (Feature Dropping)

Modelin karmaşıklığını azaltmak (overfitting'i önlemek) ve işlem hızını artırmak amacıyla bazı dönüşümler yapılmıştır:

- **Düşük Varyanslı Sütunların Atılması:** Neredeyse tüm satırlarda aynı değeri alan (örneğin Street veya Utilities) değişkenler çıkarılmıştır. Bu değişkenler model için bir ayırt edicilik sağlamadığı gibi gürültü yaratmaktadır.
- **Yüksek Eksik Veri Oranı:** Verilerin %20'sinden fazlası eksik olan sütunlar (Alley, Pool QC, Fence vb.) tamamen silinmiştir. Çok az veriye dayanarak bu sütunları doldurmak, modele yanlış bilgi verme riskini taşıdığı için bu dönüşüm tercih edilmiştir.

```
import os

# Gereksiz sütunları tanımla
gereksiz_sutunlar = [
    'Order', 'PID', 'Street', 'Alley', 'Utilities', 'Land Slope',
    'Condition 2', 'Roof Matl', 'BsmtFin SF 2', 'Heating',
    'Low Qual Fin SF', 'Bsmt Half Bath', 'Kitchen AbvGr',
    'Garage Cond', '3Ssn Porch', 'Pool Area', 'Pool QC',
    'Fence', 'Misc Feature', 'Misc Val'
]

# Bu sütunları df üzerinden çıkar
# errors='ignore' parametresi ile listedeki olmayan sütunlar hata vermeden atlanır.
df = df.drop(columns=gereksiz_sutunlar, errors='ignore')
```

2.6.2 Gereksiz Değişkenlerin Çıkarılması

2.6.3. Korelasyon Temelli Dönüşümler

Eksik veri matrisi üzerinden yapılan korelasyon analizi, bazı değişkenlerin birbiriyle yüksek oranda bağımlı olduğunu göstermiştir.

- **Neden:** Örneğin, bir evin garajı yoksa garajla ilgili 5-6 farklı sütun (Garage Type, Garage Finish, Garage Condvb.) aynı anda boştur. Bu durum, veri setinde çok fazla birbirini tekrar eden bilgi (multicollinearity) olduğunu doğrular. Bu dönüşümle, modelin bu tekrarlardan dolayı bazı özelliklere aşırı ağırlık vermesi engellenmiştir.

```
# Toplam Banyo Sayısı (Yarım banyolar 0.5 olarak eklenir)
df['TotalBathrooms'] = df['Full Bath'] + (0.5 * df['Half Bath']) + df['Bsmt Full Bath']

# Toplam Yaşam Alanı (Bodrum + Katlar)
df['TotalSF'] = df['Total Bsmt SF'] + df['Gr Liv Area']

# Evin Yaşı (Satış yılından yapım yılını çıkarıyoruz)
df['HouseAge'] = df['Yr Sold'] - df['Year Built']

# Restorasyon üzerinden geçen süre
df['RemodAge'] = df['Yr Sold'] - df['Year Remod/Add']

print("✅ Toplam banyo, alan ve yaş gibi yeni sütunlar oluşturuldu.")
```

2.6.3 Bayno için dönüşümler

2.6.4. Kategorik Verilerin Hazırlanması (Encoding Hazırlığı)

Kod içerisinde object tipindeki (metinsel) veriler, modelleme aşamasına girmeden önce analiz edilmiştir.

- **Neden:** Makine öğrenmesi algoritmaları doğrudan metinlerle işlem yapamaz. Bu yüzden MS Zoning, Neighborhood gibi değişkenlerin benzersiz değer sayıları (nunique) incelenerek, bunların ilerleyen aşamalarda sayısal temsilcilere (One-Hot veya Label Encoding) dönüştürülmesi için zemin hazırlanmıştır.

```
import pandas as pd

# Tanımlanmış kategorik sütunlar yerine, mevcut DataFrame'deki tüm 'object' türündeki sütunları alalım
categorical_cols_to_encode = df.select_dtypes(include=['object']).columns.tolist()

# 'Neighborhood_Group' özel olarak kontrol ediliyor, eğer varsa ve 'object' ise listeye eklenir.
# Zaten df.select_dtypes(include=['object']) ile otomatik alınmalı.

# One-Hot Encoding uygulamadan önce df'i kontrol edelim
# print(f"One-Hot Encoding öncesi df.info():\n{df.info()}")

# Get_dummies ile dönüşüm (ilk sütunu düşürmek multicollinearity'yi önler)
if categorical_cols_to_encode:
    print(f"✅ One-Hot Encoding uygulanacak sütunlar: {categorical_cols_to_encode}")
    df = pd.get_dummies(df, columns=categorical_cols_to_encode, drop_first=True, dtype=int) # dtype=int ekleyerek boolean yerine 0/1
    print(f"✅ Kategorik sütunlar One-Hot Encoding ile dönüştürüldü.")
else:
    print("⚠️ Dönüştürülecek 'object' tipinde kategorik sütun bulunamadı.")

# One-Hot Encoding sonrası df'i kontrol edelim
# print(f"One-Hot Encoding sonrası df.info():\n{df.info()}")

...
✅ One-Hot Encoding uygulanacak sütunlar: ['MS Zoning', 'Lot Shape', 'Land Contour', 'Lot Config', 'Condition 1', 'Bldg Type', 'House']
✅ Kategorik sütunlar One-Hot Encoding ile dönüştürüldü.
```

2.6.4 One-Hot Encoding dönüşümü

2.6.5. Logaritmik Dönüşüm Potansiyeli (SalePrice Analizi)

İstatistiksel özetlerde (df.describe()), hedef değişken olan SalePrice'ın maksimum değeri (755,000) ile ortalaması (180,796) arasında büyük bir fark olduğu görülmektedir.

- **Neden:** Bu tür sağa çarpık (skewed) dağılımlarda, fiyat farklarının model üzerindeki etkisini normalleştirmek için genellikle logaritmik dönüşüm uygulanır. Bu, yüksek fiyatlı evlerin yarattığı "aykırı değer" etkisini minimize eder.

```
import numpy as np

# Hedef değişken (SalePrice) ve alan sütunlarına Log(1+x) uyguluyoruz
log_cols = ['SalePrice', 'Lot Area', 'Gr Liv Area', 'Total Bsmt SF', '1st Flr SF']

for col in log_cols:
    df[col] = np.log1p(df[col])

print("✅ SalePrice ve temel alan sütunlarına log dönüşümü uygulandı.")

✅ SalePrice ve temel alan sütunlarına log dönüşümü uygulandı.
```

2.6.5. Logaritmik Dönüşüm

3. Makine Öğrenmesi Uygulaması

Bu aşamada, konut fiyatlarını tahmin etmek amacıyla geniş bir algoritma yelpazesi test edilmiştir. Modellerin performansı, verinin genelleme yeteneğini ölçmek için **5-Katlı Çapraz Doğrulama (5-Fold Cross Validation)** yöntemiyle değerlendirilmiştir.

3.1 Eğitilen Modeller ve Kullanılan Teknikler

Analiz sürecinde aşağıdaki modeller eğitilmiştir:

- **Doğrusal Modeller:** Linear Regression, Lasso, Ridge.
- **Komşuluk ve Ağaç Temelli Modeller:** KNN, Decision Tree, Random Forest.
- **Boosting Algoritmaları:** XGBoost, LightGBM, CatBoost.
- **Yapay Sinir Ağları:** MLP (1, 2 ve 3 katmanlı).
- **Topluluk (Ensemble) Modeli:** Linear Regression, RF ve XGBoost modellerinin birleşiminden oluşan VotingRegressor.

3.2 Eğitim ve Test Sonuçları

Modellerin performans metrikleri (RMSE, MAE, vb.) karşılaştırıldığında öne çıkan sonuçlar şöyledir:

	Model	MSE	RMSE	MAE	MAPE	R2
0	LinearRegression	0.018922	0.137558	0.084676	0.007110	0.886058
1	Lasso	0.041856	0.204588	0.136437	0.011432	0.747956
2	Ridge	0.016260	0.127514	0.082166	0.006891	0.902090
3	SVR	0.037131	0.192695	0.130418	0.010960	0.776409
4	KNN	0.049001	0.221362	0.154417	0.012929	0.704934
5	DecisionTree	0.038753	0.196858	0.135720	0.011385	0.766643
6	RandomForest	0.019768	0.140597	0.091468	0.007679	0.880967
7	XGBoost	0.017912	0.133836	0.090447	0.007596	0.892140
8	CatBoost	0.014010	0.118363	0.076181	0.006404	0.915638
9	LightGBM	0.016143	0.127056	0.083293	0.007001	0.902791
10	MLP_1layer	9.755166	3.123326	1.888530	0.156663	-57.741975
11	MLP_2layer	5.930550	2.435272	1.383264	0.114765	-34.711561
12	MLP_3layer	1.129480	1.062770	0.636833	0.053109	-5.801307
13	Ensemble	0.015392	0.124065	0.080010	0.006721	0.907315

3.2 Modellerin sonuçları

Yorum: En yüksek başarıyı skoru ile **CatBoost** algoritması göstermiştir. Boosting modellerinin (CatBoost, LightGBM) karmaşık ve doğrusal olmayan ilişkileri yakalamada doğrusal modellere göre daha üstün olduğu görülmüştür. MLP modelleri başlangıç parametreleriyle negatif sonuçlar vererek düşük performans sergilemiştir.

3.3 Hiperparametre ayarına aktarılan en iyi 3 model:

1. **CatBoost**
2. **LightGBM**
3. **MLP_2layer** (Potansiyelini görmek adına optimizasyon aşamasına dahil edilmiştir).

4. Hiperparametre Optimizasyonu

Seçilen 3 model üzerinde ParameterGrid yöntemi kullanılarak en iyi parametre kombinasyonları aranmıştır. Her model için en az 2 parametre ve 3 seviye kuralına uygun olarak şu aralıklar test edilmiştir:

- **LightGBM:**
 - n_estimators: [100, 200]
 - max_depth: [5, 10]
- **CatBoost:**
 - iterations: [100, 200]
 - depth: [4, 6]
- **MLP (Yapay Sinir Ağları):**
 - hidden_layer_sizes: [(64, 32), (128, 64)]
 - alpha (Düzenleştirme): [0.001, 0.01]

Sonuç: Yapılan denemeler sonucunda, hem eğitim süresi hem de doğrulama verisi üzerindeki kararlılığı göz önüne alınarak **LightGBM** (Parametreler: max_depth=10, n_estimators=100) final modeli olarak seçilmiştir.

5. Modelin Açıklanması (SHAP)

Modelin kararlarını şeffaf hale getirmek için **SHAP (SHapley Additive exPlanations)** analizi uygulanmıştır.

- **SHAP Summary Plot Analizi:** Grafik incelendiğinde, fiyatı etkileyen en kritik özelliklerin **Overall Qual** (Genel Kalite), **Gr Liv Area** (Yaşam Alanı Metrekaresi) ve **TotalSF** (Toplam Alan) olduğu görülmektedir.




5.1 SHAP

Yorum: Kalite puanı arttıkça ve yaşam alanı büyüdükçe modelin tahmin ettiği fiyat pozitif yönde (kırmızı renkler sağ tarafta) artmaktadır. Evin yaşı (**HouseAge**) ise genellikle fiyat üzerinde negatif bir etkiye sahiptir; ev yaşlandıkça SHAP değerleri negatif yöne kayarak fiyat tahminini düşürmektedir.

6. Arayüz Tasarımı

Projenin son aşamasında, kullanıcıların ev özelliklerini girerek anlık tahmin alabileceği bir **Gradio** arayüzü tasarlanmıştır.

- **Girdiler:** Kullanıcı; Arsa Alanı, Yaşam Alanı, Toplam Kat Alanı, Genel Kalite (1-10), Evin Yaşı, Banyo Sayısı ve Mahalle Grubu gibi parametreleri slider ve dropdown menüler aracılığıyla seçebilmektedir.
- **İşleyiş:** Arayüz, girilen değerleri modelin eğitildiği formata (log dönüşümleri ve One-Hot encoding dahil) getirir ve eğitilen en iyi model üzerinden tahmin yürütür.
- **Çıktılar:**
 1. **Tahmin Edilen Değer:** Evin piyasa değeri (Örn: \$245,500.00).
 2. **SHAP Waterfall Plot:** O spesifik ev tahmini için hangi özelliğin fiyatı ne kadar yukarı çektiğini veya aşağı indirdiğini gösteren görsel açıklama.

 **Ames Housing: Mükemmel Ev Fiyat Tahmini**

Evin özelliklerini girin ve modelin fiyatı belirlerken hangi özelliklere (SHAP) ne kadar önem verdiğini görün.

Arsa Alanı (Lot Area)

500

500

20000

Yaşam Alanı (Gr Liv Area)

500

500

5000

Toplam Kat Alanı (TotalSF)

500

500

8000

Genel Kalite (Overall Qual)

1

1

10

Evin Yaşı (HouseAge)

0

0

150

Toplam Banyo Sayısı (TotalBathrooms)

1

1

6

Mahalle Grubu

Budget

Kalite Seviyesi

Low

Sonuç

SHAP Analizi (Neden Bu Fiyat?)

Flag

Clear

Submit

Use via API · Built with Gradio · Settings

6.1 Modelin Arayüzü

Bu tasarım, teknik olmayan kullanıcıların bile modelin arkasındaki mantığı anlamasını ve veriye dayalı kararlar almasını sağlamaktadır.

14