

Ondokuz Mayıs Üniversitesi

Bilgisayar mühendisliği



Hami Keskin

17060280

Regresyon modeli ile fiyat tahmini

Zinnet Duygu Akşehir

```
# Evin yaşı, yatak odası sayısı, alan nüfusu vb.
# kullanarak ev fiyatlarını tahmin etmek için klasik Lineer regresyon kullanacağız.
```

```
# Kütüphaneler
```

```
# Veriyi yüklemek ve analiz yapmak için pandas ve numpy
import pandas as pd
import numpy as np
```

```
import matplotlib.pyplot as plt
```

```
# Lineer regresyon, hata skorlarının kullanılması için
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score, mean_absolute_error, mean_squared_error
```

```
# Veriyi yüklüyoruz
```

```
df = pd.read_csv("USA_Housing.csv");
```

```
# Veri setinin ilk 5 değerine bakalım
```

```
df.head()
```

	Avg. Area Income	Avg. Area House Age	Avg. Area Number of Rooms	Avg. Area Number of Bedrooms	Area Population	Price	Address
0	79545.458574	5.682861	7.009188	4.09	23086.800503	1.059034e+06	208 Michael Ferry Apt. 674\nLaurabury, NE 3701...
1	79248.642455	6.002900	6.730821	3.09	40173.072174	1.505891e+06	188 Johnson Views Suite 079\nLake Kathleen, CA...
2	61287.067179	5.865890	8.512727	5.13	36882.159400	1.058988e+06	9127 Elizabeth Stravenue\nDanieltown, WI 06482...
3	63345.240046	7.188236	5.586729	3.26	34310.242831	1.260617e+06	USS Barnett\nFPO AP 44820
4	59982.197226	5.040555	7.839388	4.23	26354.109472	6.309435e+05	USNS Raymond\nFPO AE 09386

```
# Veri seti Amerika'da bulunan evlerin bilgisini içeriyor.
```

```
# Veri kümesinde adres dışındaki tüm sütunların sayı olduğunu görebiliyoruz,
# bunu df.info() kullanarak bulabiliriz:
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 5000 entries, 0 to 4999
```

```
Data columns (total 7 columns):
```

#	Column	Non-Null Count	Dtype
0	Avg. Area Income	5000 non-null	float64
1	Avg. Area House Age	5000 non-null	float64
2	Avg. Area Number of Rooms	5000 non-null	float64
3	Avg. Area Number of Bedrooms	5000 non-null	float64
4	Area Population	5000 non-null	float64
5	Price	5000 non-null	float64
6	Address	5000 non-null	object

```
dtypes: float64(6), object(1)
```

```
memory usage: 273.6+ KB
```

```
# Adres sütununu string tipinde olduğundan çıkaralım,
```

```
# çünkü bu modelde bu veriyle çalışamayız.
```

```
df = df.drop("Address", axis=1)
```

```
# Verimizde boş değer var mı diye kontrol edelim.
```

```
df.isnull().sum()
```

```
# Verimizde boş değer yok.
```

```
Avg. Area Income      0
Avg. Area House Age    0
Avg. Area Number of Rooms  0
Avg. Area Number of Bedrooms  0
Area Population         0
Price                  0
dtype: int64
```

```
# Veriyi X ve y olmak iki değişkene bölelim

# X, tahmin etmeye çalıştığımız sütun, fiyat sütunu olmadan veri kümesini içerir
X = df.drop("Price", axis=1)

# y değişkeni, tahmin etmeye çalıştığımız fiyat sütununu içerir.
y = df["Price"]

# Veri setini test ve eğitim için bölelim
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size= 0.2, random_state= 101)
# verinin %80'i eğitim için ayırdık

ln = LinearRegression()

# Alttaki kodda eğitim setimizi eğitiyoruz.
ln.fit(X_train,y_train)

LinearRegression()

# Eğitim sonucu bulduğumuz fonksiyonun çıktısını alalım.
print("Elde edilen regresyon modeli: Y={}+{}X".format(ln.intercept_, ln.coef_[0]))

Elde edilen regresyon modeli: Y=-2640441.399782746+21.566695744660596X

# Şimdi regresyonun performansına bakalım.
y_predicted = ln.predict(X)
r2_score(y,y_predicted)

0.9180061818411438
```