BLM4510 Yapay Zeka
Ödev 2

Hazırlayanlar:

Melih Ocakcı 19011061
Onur Demir 18011078

Veri Kümesi:

```
total_page_web_element = driver.find_element_by_xpath('//*[@id="searchResultsSearchForm"]/div/div[4]/div[3]/p')
```

Veriyi toplarken sahibinden.com dan selenium yazılımı ile dom objelerini topladık. Veriyi toplarken 3 tabloya baktık.

349.500 TL Kredi Teklifi Al

İstanbul / Bayrampaşa / Yıldırım Mh.

İlan No	1028233452
İlan Tarihi	25 Mayıs 2022
Marka	BMW
Seri	3 Serisi
Model	320d Premium
Yıl	2010
Yakıt	Dizel
Vites	Otomatik
KM	276.000
Kasa Tipi	Sedan
Motor Gücü	177 hp
Motor Hacmi	1995 cc
Çekiş	Arkadan İtiş
Renk	Gri
Garanti	Hayır
Plaka / Uyruk	Türkiye (TR) Plakalı
Kimden	Sahibinden
Görüntülü Arama İle Görülebilir	Hayır
Takas	Evet
Durumu	İkinci El

İlk tablomuz bu şekildedir.

İlan Detayları | Teknik Özellikler

3 Serisi 320d Premium

Genel Bakış

Araç Tipi	Binek Araç / D Segment
Kasa Tipi / Kapı Sayısı	Sedan / 4 Kapı
Motor Tipi	Dizel / 4 silindir
Üretim Yılı (İlk / Son)	2008 / 2012
Yakıt Tüketimi (Şehir içi / Şehir dışı)	6 lt / 4,1 lt
Motor Gücü	177 hp
Şanzıman	Otomatik / 6 Vites / Arkadan İtiş
Hızlanma 0-100 km/saat	8,3 sn

Ikinci tablomuz şu şekildedir.

Donanım	Standart Donanım	Opsiyonel Donanım
Güvenlik		
Arka Perde Hava Yastığı	✔	
Disk Frenler	✔	
Dizel Partikül Filtresi	✔	
Entegre Kemer Gergi Sistemi	✔	
Isofix Çocuk Koltuğu Bağlantısı	✔	
Ön Yolcu Hava Yastığı	✔	

Üçüncü tablomuz da donabnım hakkındaki şu tablodur.

Bunları toplarken 600 satır 300 sütün veri elde ettik.

Tramer kaydını alırken regex yaklaşımı kullandık.Çoğu kişi tramer ve benzeri kelimelerle tramer kaydının bilgisini girmiştir.Aşağıda bunun kodu görülmektedir.

```
hasTramerRegister = re.search(r'( TL)|(TL )', ad_detail_text, flags=re.IGNORECASE)
if(hasTramerRegister == None):
    tramer_info = re.findall(r'((tramer|hasar).(kayd[ıi])?.(\w+))',ad_detail_text, flags=re.IGNORECASE)
    if tramer_info != []:
        for string in tramer_info[0]:
            if(bool(re.search('var',string, re.IGNORECASE)) or bool(re.search('mevcut',string, re.IGNORECASE))):
                hasTramerRegister = True
        if(hasTramerRegister == None):
            hasTramerRegister = False
    else:
        # ek işlev düşünülecek
        # tramer_info = re.findall(r'(YOKTUR)',ad_detail_text, flags=re.IGNORECASE)
        hasTramerRegister = False
else:
    hasTramerRegister = True
```

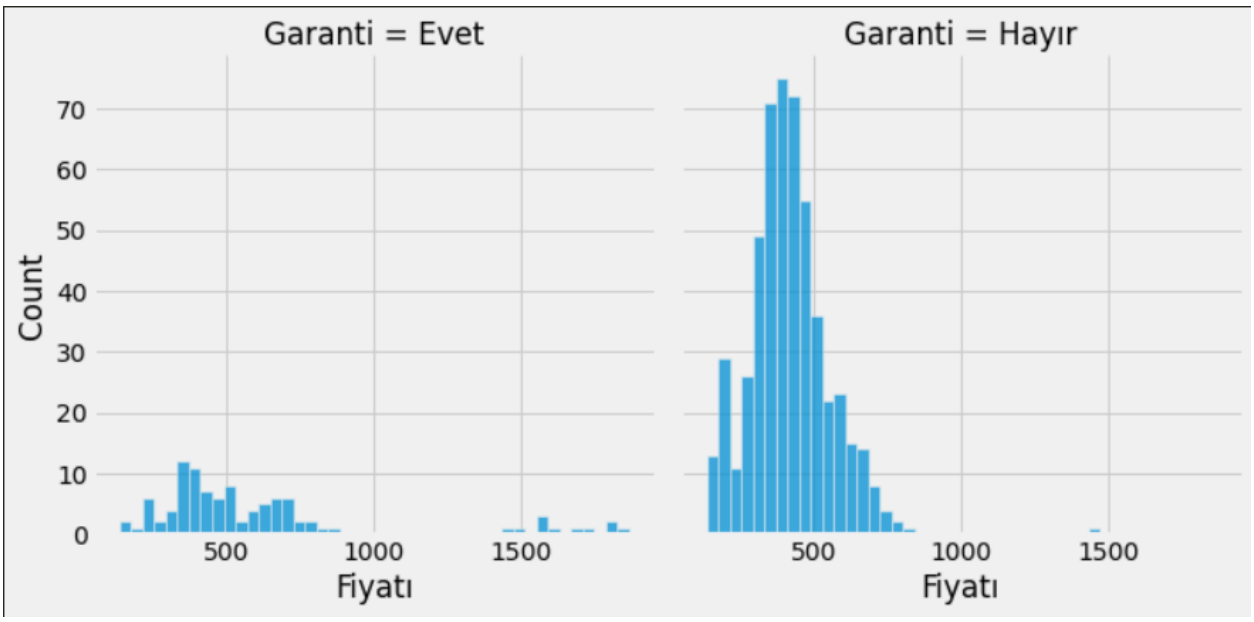
Ön İşleme:

Verinin ham halinde her bir araç için 300 adet özellik bulunmaktaydı. Bu verilerin çoğu modele verilmeden önce kaldırıldı. Kullanılan özelliklerin listesi aşağıdaki gibidir.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 626 entries, 0 to 625
Data columns (total 17 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Hasar Kaydı             626 non-null    int64
 1   KM                      626 non-null    float64
 2   Garanti                 626 non-null    object
 3   Vites                   626 non-null    object
 4   Çekiş                   626 non-null    object
 5   Fiyatı                  626 non-null    float64
 6   Motor Tipi              626 non-null    object
 7   Üretim Yılı             626 non-null    object
 8   Motor Gücü              626 non-null    object
 9   Şanzıman                626 non-null    object
 10  Hızlanma 0-100 km/saat  626 non-null    object
 11  Azami Sürat             626 non-null    object
 12  Motor Hacmi             626 non-null    object
 13  Yakıt Tipi              626 non-null    object
 14  Ortalama                626 non-null    object
 15  Durumu                  626 non-null    object
 16  Kimden                  626 non-null    object
dtypes: float64(2), int64(1), object(14)
memory usage: 83.3+ KB
```

Sütunları filtreleme aşamasında PCA gibi metodlar kullanmayı denesek de sonuçta kendi bilgimizden yola çıkarak uygun olan sütunları seçtik. Görüldüğü gibi sütunların hiçbirinde null değeri yoktur dolayısıyla ayıklama gerekli değildir.

Bu sütunlardan “Fiyatı” isimli sütun hedef sütundur. Veri seti programa verilmeden önce sayısal değere çevrilmiştir. Bunun yanında modelin diğer değişkenleri işleyebilmesi için tüm kategorik veriler OrdinalEncoder fonksiyonu ile sayısal değerlere çevrilmiştir.

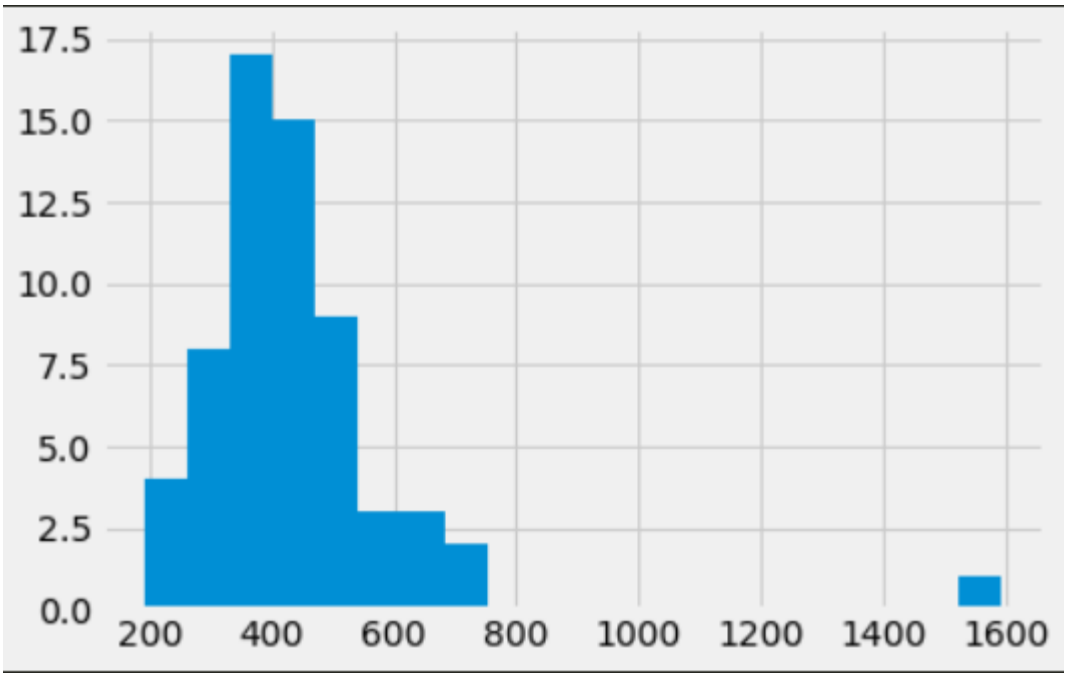


Her bir sütünun dağılımı inceledik ve buna göre kullanıcağımız sütünları belirledik.

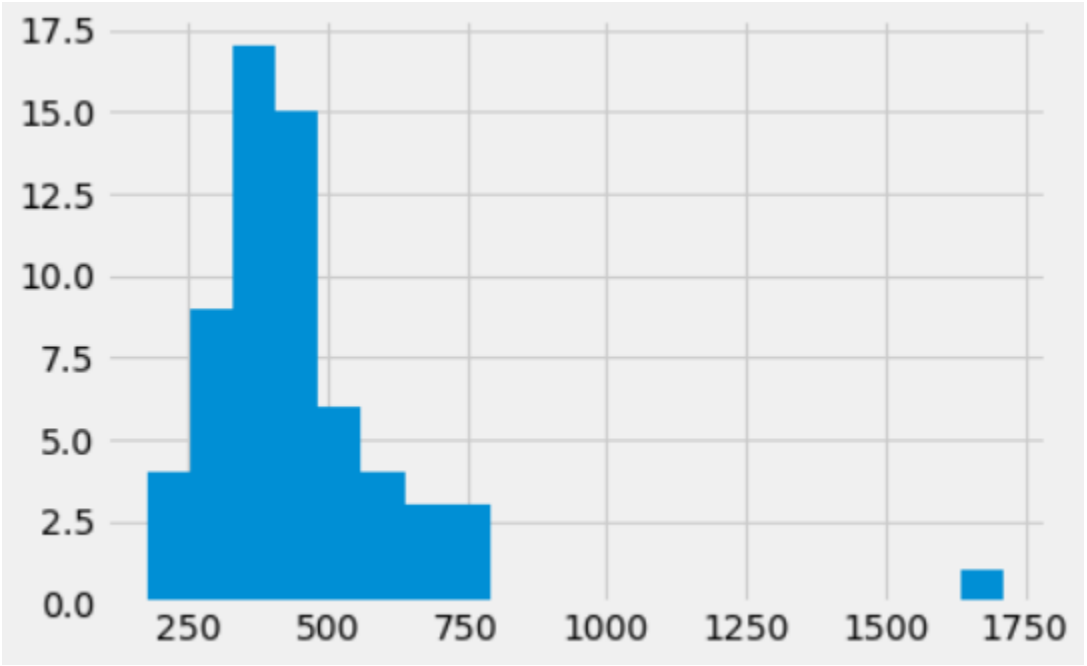
Modellerin Eğitilmesi:

Bu projede 5 adet regresyon algoritmasına yer verilmiştir. Bunlar support vector regression, linear regression, decision tree regression, k-neighbors regression ve random forest regression algoritmalarıdır.

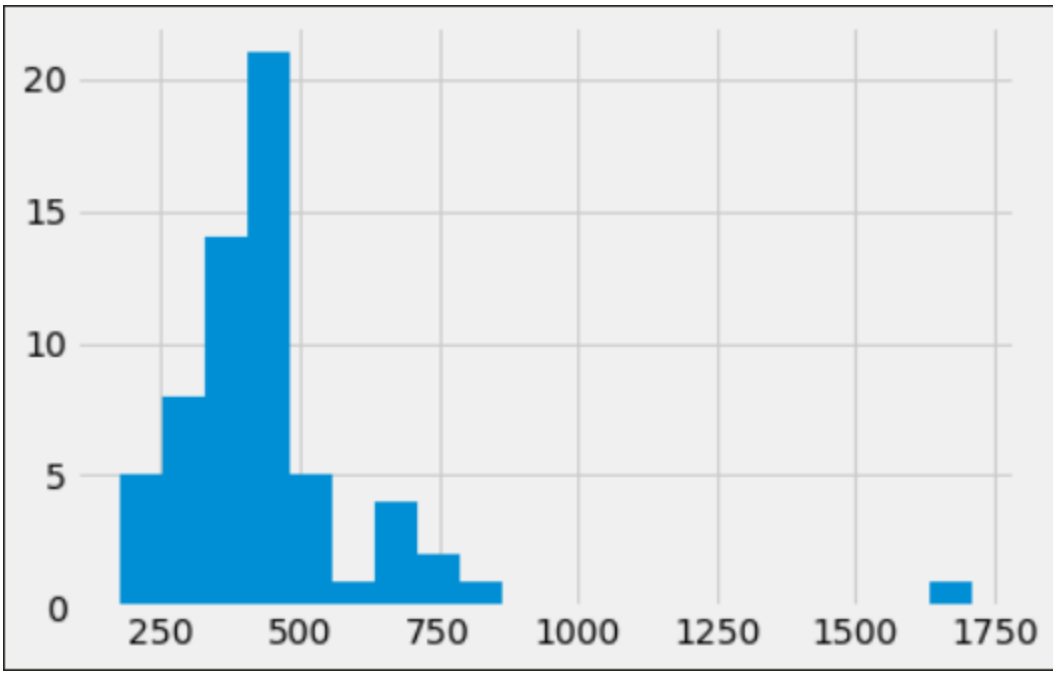
Algoritmaları kıyaslamak için daha güvenilir olması sebebiyle 10 katlı k-fold cross validation kullanılmıştır.



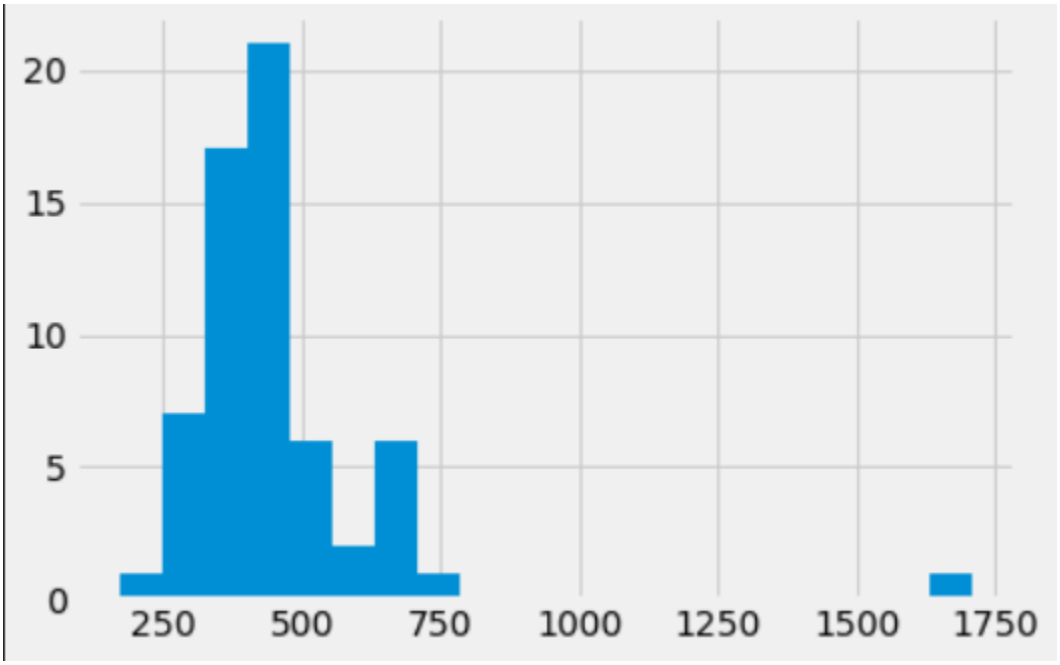
Lineer Regresyon Histogram Çıktısı



Decision Tree Histogram Çıktısı

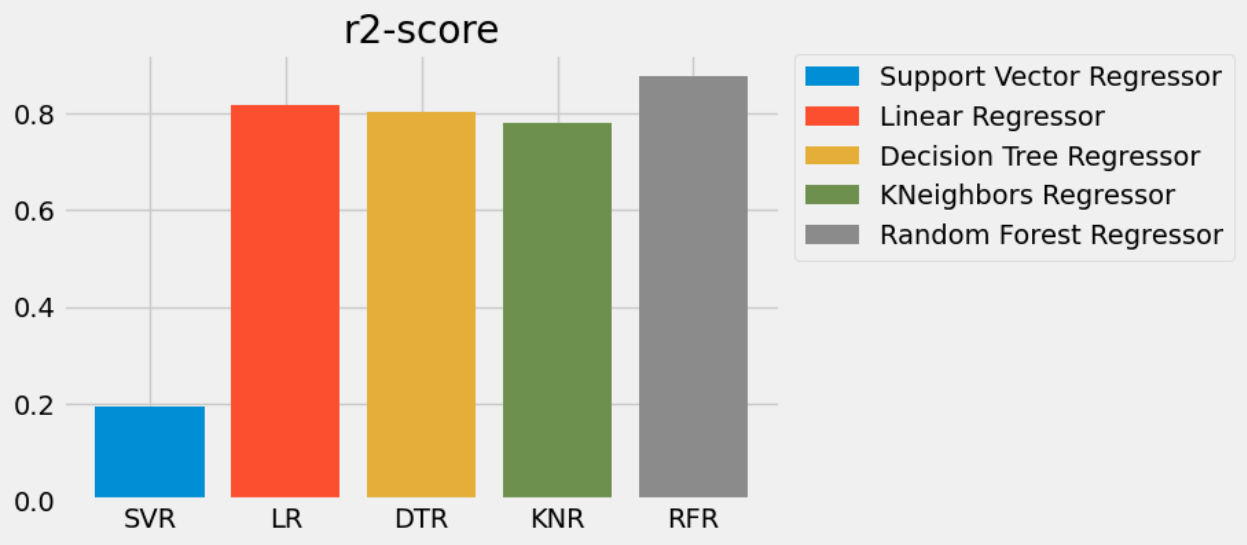


Random Forest Histogram Çıktısı



K-Nearest Neighbors Histogram Çıktısı

Modellerin Değerlendirilmesi:



Modellerin Ortalama r2-score Değerleri

Yukarıdaki şekilde her bir model için 10 katlı cross validation r2-score değerlerinin ortalamaları görülmektedir. Görüldüğü üzere en yüksek değere random forest algoritması sahipken SVR algoritması açık ara en kötü performansı vermiştir.

Modellerin çıktılarını t-test kullanarak kıyasladık ve istatistiksel olarak farklı olduklarını gördük.

```
print(stats.ttest_ind(KNREG, DT))
print(stats.ttest_ind(RFREG, DT))
print(stats.ttest_ind(LR, LR))
print(stats.ttest_ind(KNREG, RFREG))
print(stats.ttest_ind(LR, KNREG))
```

```
Ttest_indResult(statistic=0.2006041270776536, pvalue=0.8413421668005895)
Ttest_indResult(statistic=-0.037528619861820704, pvalue=0.9701248600585546)
Ttest_indResult(statistic=0.0, pvalue=1.0)
Ttest_indResult(statistic=0.23687579445101092, pvalue=0.8131504297004774)
Ttest_indResult(statistic=-0.40380789022601776, pvalue=0.6870609237691514)
```

Farklı modeller için t-test sonuçları

Testlerin sonucunda decision tree algoritması ile random forest algoritmasının çok benzer sonuçlara sahip olduğunu gördük. Bunun dışında diğer algoritmalar birbirinden olduğunca farklı sonuçlar verdi.