

pusula-muharrem-gedik

September 22, 2024

Muharrem Gedik gdk.muharrem@gmail.com

```
[1]: import pandas as pd

# Excel dosyasını okuma
df = pd.read_excel('side_effect_data 1.xlsx', sheet_name='Sheet1')
# Veri türünü kontrol etme
print(type(df))
```

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

```
[2]: # Veri çerçevesinin ilk 10 satırını görüntüleme
df.head(10)
```

```
[2]:
```

	Kullanici_id	Cinsiyet	Dogum_Tarihi	Uyruk	Il \
0	107	Male	1960-03-01	Turkiye	Canakkale
1	140	Male	1939-10-12	Turkiye	Trabzon
2	2	Female	1976-12-17	Turkiye	Canakkale
3	83	Male	1977-06-17	Turkiye	Adana
4	7	Female	1976-09-03	Turkiye	Izmir
5	131	Male	1982-01-05	Turkiye	NaN
6	61	NaN	1997-01-10	Turkiye	Mersin
7	95	NaN	1997-01-15	Turkiye	Antalya
8	3	Female	1973-08-05	Turkiye	NaN
9	53	NaN	1941-10-16	Turkiye	Adana

	Ilac_Adi	Ilac_Baslangic_Tarihi	Ilac_Bitis_Tarihi \
0	trifluoperazine	2022-01-09	2022-03-04
1	fluphenazine hcl	2022-01-09	2022-03-08
2	warfarin sodium	2022-01-11	2022-03-12
3	valproic acid	2022-01-04	2022-03-12
4	carbamazepine extended release	2022-01-13	2022-03-06
5	fluoxetine dr	2022-01-07	2022-03-19
6	carbamazepine extended release	2022-01-05	2022-03-07
7	thiothixene	2022-01-10	2022-03-07
8	asenapine tablet, sublingual	2022-01-08	2022-03-17
9	fluphenazine hcl elixir	2022-01-10	2022-03-13

	Yan_Etki	Yan_Etki_Bildirim_Tarihi	Alerjilerim	\
0	Kabizlik	2022-02-19 18:28:43	Ceviz	
1	Yorgunluk	2022-02-03 20:48:17	Toz	
2	Carpinti	2022-02-04 05:29:20	Muz	
3	Sinirlilik	2022-02-08 01:01:21	Pancar	
4	Agizda Farkli Bir Tat	2022-02-12 05:33:06	NaN	
5	Agizda Farkli Bir Tat	2022-02-04 08:08:31	NaN	
6	Agizda Farkli Bir Tat	2022-02-19 17:39:48	Bugday	
7	Deride Morarma	2022-02-09 20:53:54	Ispanak	
8	Bas Agrisi	2022-02-19 04:14:26	Laktoz	
9	Gormede Bulaniklik	2022-02-09 01:16:50	Elma	

	Kronik Hastaliklarim	Baba Kronik Hastaliklari	\
0	Hipertansiyon, Kan Hastaliklari	Guatr, Hipertansiyon	
1	NaN	Guatr, Diger	
2	Kalp Hastaliklari, Diyabet	Diyabet, KOAH	
3	Diyabet, Diger	Kalp Hastaliklari, Diger	
4	Diyabet, Kalp Hastaliklari	Alzheimer, Hipertansiyon	
5	Diyabet, Hipertansiyon	NaN	
6	KOAH, Diyabet	Kalp Hastaliklari, Kan Hastaliklari	
7	Astim, Diyabet	Alzheimer, Diyabet	
8	NaN	Kan Hastaliklari, Diger	
9	Kemik Erimesi, KOAH	Hipertansiyon	

	Anne Kronik Hastaliklari	Kiz Kardes Kronik Hastaliklari	\
0	KOAH	Kemik Erimesi, Kalp Hastaliklari	
1	Hipertansiyon, Kalp Hastaliklari		
2	Kemik Erimesi, Diyabet	Diyabet, Kemik Erimesi	
3	NaN	Astim	
4	Kan Hastaliklari, Kemik Erimesi	Diyabet, Diger	
5	Alzheimer, Kalp Hastaliklari	Kanser	
6	NaN	Kan Hastaliklari, Hipertansiyon	
7	KOAH	Guatr, Kan Hastaliklari	
8	Kanser, Guatr	Alzheimer, Hipertansiyon	
9	KOAH, Alzheimer	KOAH, Alzheimer	

	Erkek Kardes Kronik Hastaliklari	Kan Grubu	Kilo	Boy
0	Kemik Erimesi, Guatr	B RH-	103.0	191.0
1	KOAH, Diyabet	NaN	81.0	181.0
2	Diger	B RH-	93.0	158.0
3	Kalp Hastaliklari, Kanser	AB RH-	NaN	165.0
4	Alzheimer, Hipertansiyon	AB RH-	99.0	172.0
5	Diyabet, Kemik Erimesi	AB RH-	66.0	150.0
6	KOAH, Kalp Hastaliklari	NaN	NaN	147.0
7	Guatr, Kan Hastaliklari	B RH+	96.0	162.0
8	Kanser, Kalp Hastaliklari	O RH-	91.0	NaN
9	Astim, Kan Hastaliklari	O RH-	107.0	156.0

```
[3]: # Veri çerçevesinin son 10 satırını görüntüleme
df.tail(10)
```

```
[3]:      Kullanici_id Cinsiyet Dogum_Tarihi    Uyruk      Il \
2347           148     Male   1981-08-31  Turkiye    Bursa
2348           112      NaN   1950-02-03  Turkiye  Trabzon
2349            9      NaN   1957-01-04  Turkiye     NaN
2350          153      NaN   1975-05-19  Turkiye    Adana
2351          104   Female   1967-07-11  Turkiye  Istanbul
2352            9      NaN   1957-01-04  Turkiye     NaN
2353          101   Female   2004-11-09  Turkiye    Mersin
2354          127   Female   1951-11-29  Turkiye    Mersin
2355          178     Male   1980-01-30  Turkiye  Kayseri
2356          174   Female   1986-11-07  Turkiye  Istanbul
```

```
      Ilac_Adi Ilac_Baslangic_Tarihi \
2347           iloperidone          2022-01-01
2348    lithium carbonate er          2022-01-04
2349    tranylcypromine          2022-01-14
2350    trifluoperazine          2022-01-07
2351     vortioxetine          2022-01-02
2352 desoximetasone spray, non-aerosol          2022-01-13
2353    olanzapine-fluoxetine          2022-01-02
2354           trazodone          2022-01-02
2355    duloxetine hydrochloride          2022-01-02
2356           valproic acid          2022-01-06
```

```
      Ilac_Bitis_Tarihi      Yan_Etki Yan_Etki_Bildirim_Tarihi \
2347      2022-03-16      Yorgunluk      2022-02-03 20:48:17
2348      2022-03-18      Yorgunluk      2022-02-03 20:48:17
2349      2022-03-09      Tansiyon Yükselme      2022-02-08 07:55:59
2350      2022-03-16      Istah Artisi      2022-02-17 07:08:01
2351      2022-03-07      Ishal      2022-02-12 19:13:43
2352      2022-03-04      Ishal      2022-02-12 19:13:43
2353      2022-03-05      Agizda Farkli Bir Tat      2022-02-19 17:39:48
2354      2022-03-12      Yorgunluk      2022-02-03 20:48:17
2355      2022-03-08      Carpinti      2022-02-04 05:29:20
2356      2022-03-06      Istah Artisi      2022-02-17 07:08:01
```

```
      Alerjilerim      Kronik Hastaliklarim \
2347      Ispanak      Kemik Erimesi, Kan Hastaliklari
2348      NaN      Kan Hastaliklari, Hipertansiyon
2349      Ispanakgiller      NaN
2350      Deniz Urunler      Hipertansiyon, Kalp Hastaliklari
2351      Kahve      KOAH, Hipertansiyon
2352      Ispanakgiller      NaN
2353      Istiridye      NaN
```

2354	Deniz Urunler	Guatr, KOAH	
2355	Sari Kantaron	Alzheimer, Diger	
2356	NaN	Alzheimer, Diger	
	Baba Kronik Hastaliklari	Anne Kronik Hastaliklari \	
2347	Astim, Kanser		
2348	Kalp Hastaliklari, Kanser	Kalp Hastaliklari, Diyabet	
2349		Astim, Guatr	
2350	Kemik Erimesi	Astim, Hipertansiyon	
2351	Kalp Hastaliklari, Kanser	Kalp Hastaliklari, Astim	
2352		Astim, Guatr	
2353	Hipertansiyon, Astim	Astim, Kemik Erimesi	
2354	Alzheimer, Diger	NaN	
2355	NaN	Kalp Hastaliklari, Diger	
2356	Kanser, Diger	Hipertansiyon, Kan Hastaliklari	
	Kiz Kardes Kronik Hastaliklari	Erkek Kardes Kronik Hastaliklari \	
2347	Alzheimer	Kalp Hastaliklari, Diyabet	
2348	Guatr, Diger	Diger	
2349		KOAH, Kan Hastaliklari	
2350	Diyabet, Astim	Diyabet, Kemik Erimesi	
2351	Kalp Hastaliklari, Astim	Diyabet, Astim	
2352		KOAH, Kan Hastaliklari	
2353	KOAH, Kan Hastaliklari	Astim, Diyabet	
2354	Astim, Diyabet	Kalp Hastaliklari, Diger	
2355	Kanser, KOAH	Astim, KOAH	
2356		Hipertansiyon, Diger	
	Kan Grubu	Kilo	Boy
2347	0 RH+	94.0	NaN
2348	0 RH-	80.0	NaN
2349	NaN	50.0	152.0
2350	B RH-	77.0	149.0
2351	A RH-	79.0	191.0
2352	NaN	50.0	152.0
2353	AB RH-	NaN	178.0
2354	B RH+	90.0	203.0
2355	NaN	90.0	184.0
2356	AB RH+	79.0	175.0

```
[4]: # veri türleri, veri sayısı ve eksik değerler hakkında özet bilgileri sağlama
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2357 entries, 0 to 2356
Data columns (total 19 columns):
#   Column                                Non-Null Count  Dtype
```

```

---  -----
0    Kullanici_id          2357 non-null    int64
1    Cinsiyet              1579 non-null    object
2    Dogum_Tarihi          2357 non-null    datetime64[ns]
3    Uyruk                 2357 non-null    object
4    Il                    2130 non-null    object
5    Ilac_Adi              2357 non-null    object
6    Ilac_Baslangic_Tarihi 2357 non-null    datetime64[ns]
7    Ilac_Bitis_Tarihi     2357 non-null    datetime64[ns]
8    Yan_Etki              2357 non-null    object
9    Yan_Etki_Bildirim_Tarihi 2357 non-null    datetime64[ns]
10   Alerjilerim           1873 non-null    object
11   Kronik Hastaliklarim   1965 non-null    object
12   Baba Kronik Hastaliklari 2201 non-null    object
13   Anne Kronik Hastaliklari 2140 non-null    object
14   Kiz Kardes Kronik Hastaliklari 2260 non-null    object
15   Erkek Kardes Kronik Hastaliklari 2236 non-null    object
16   Kan Grubu             2010 non-null    object
17   Kilo                  2064 non-null    float64
18   Boy                   2243 non-null    float64
dtypes: datetime64[ns](4), float64(2), int64(1), object(12)
memory usage: 350.0+ KB

```

0.0.1 Object türündeki sütunları categorical türüne çevireceğiz.

Neden Object Verileri Categorical Türüne Çevirmelisiniz?

Bellek Kullanımını Optimize Etme Object türündeki veriler genellikle string (metin) verileri içerir ve bellekte daha fazla yer kaplayabilir. Pandas'ın categorical veri türü, bu metinleri daha kompakt bir şekilde temsil eder ve bellekte yer tasarrufu sağlar.

Performans İyileştirmeleri Kategorik veriler üzerinde yapılan işlemler (örneğin, gruplama, sıralama) genellikle daha hızlıdır çünkü categorical veri türü, verileri sayısal kodlara dönüştürür ve bu sayede işlem süresi azalır.

Modelleme İhtiyaçları Çoğu makine öğrenimi algoritması, sayısal verilerle çalışır. object türündeki kategorik verileri categorical türüne dönüştürmek, bu verileri sayısal formatlara dönüştürmek için daha uygun hale getirir. Ayrıca, bazı modeller bu tür verileri doğrudan işleyebilir.

Kategorik Özellikleri Daha İyi Yönetme categorical veri türü, her kategori için bir etiket ve bu etiketlerin sıralı veya sırasız olup olmadığını belirlemeye olanak tanır. Bu, veri analizi ve işleme sürecinde daha fazla kontrol sağlar.

```

[5]: # 'object' türündeki sütunları 'categorical' türüne çevirme
for col in df.select_dtypes(include='object').columns:
    df[col] = df[col].astype('category')

```

```
[6]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2357 entries, 0 to 2356
Data columns (total 19 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Kullanici_id                         2357 non-null   int64
1   Cinsiyet                             1579 non-null   category
2   Dogum_Tarihi                         2357 non-null   datetime64[ns]
3   Uyruk                                2357 non-null   category
4   Il                                    2130 non-null   category
5   Ilac_Adi                             2357 non-null   category
6   Ilac_Baslangic_Tarihi                2357 non-null   datetime64[ns]
7   Ilac_Bitis_Tarihi                   2357 non-null   datetime64[ns]
8   Yan_Etki                             2357 non-null   category
9   Yan_Etki_Bildirim_Tarihi             2357 non-null   datetime64[ns]
10  Alerjilerim                          1873 non-null   category
11  Kronik Hastaliklarim                 1965 non-null   category
12  Baba Kronik Hastaliklari             2201 non-null   category
13  Anne Kronik Hastaliklari             2140 non-null   category
14  Kiz Kardes Kronik Hastaliklari       2260 non-null   category
15  Erkek Kardes Kronik Hastaliklari     2236 non-null   category
16  Kan Grubu                            2010 non-null   category
17  Kilo                                  2064 non-null   float64
18  Boy                                   2243 non-null   float64
dtypes: category(12), datetime64[ns](4), float64(2), int64(1)
memory usage: 181.1 KB
```

```
[7]: # Satır ve sütun sayısını ekrana basma
rows, columns = df.shape
print("Satır sayısı:", rows)
print("Sütun sayısı:", columns)
```

Satır sayısı: 2357

Sütun sayısı: 19

0.0.2 Eksik verilerimiz için yapacağımız işlemlere karar vermek için veri türlerimizi derinlemesine incelemeliyiz.

İlk olarak category türündeki verileri inceleyeceğiz, benzersiz unique değerleri tespit edeceğiz.

Daha sonra sayısal değerleri ve son olarak da tarihsel değerleri inceleyeceğiz.

1. Category türündeki verilerin kategorilerini, sayısını, frekanslarını inceleyeceğiz ve görselleştirerek anlaşılabilir kılacağız.

```
[8]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2357 entries, 0 to 2356
Data columns (total 19 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Kullanici_id                          2357 non-null   int64
1   Cinsiyet                             1579 non-null   category
2   Dogum_Tarihi                          2357 non-null   datetime64[ns]
3   Uyruk                                 2357 non-null   category
4   Il                                     2130 non-null   category
5   Ilac_Adi                             2357 non-null   category
6   Ilac_Baslangic_Tarihi                 2357 non-null   datetime64[ns]
7   Ilac_Bitis_Tarihi                     2357 non-null   datetime64[ns]
8   Yan_Etki                              2357 non-null   category
9   Yan_Etki_Bildirim_Tarihi              2357 non-null   datetime64[ns]
10  Alerjilerim                           1873 non-null   category
11  Kronik Hastaliklarim                  1965 non-null   category
12  Baba Kronik Hastaliklari              2201 non-null   category
13  Anne Kronik Hastaliklari              2140 non-null   category
14  Kiz Kardes Kronik Hastaliklari        2260 non-null   category
15  Erkek Kardes Kronik Hastaliklari      2236 non-null   category
16  Kan Grubu                             2010 non-null   category
17  Kilo                                  2064 non-null   float64
18  Boy                                    2243 non-null   float64
dtypes: category(12), datetime64[ns](4), float64(2), int64(1)
memory usage: 181.1 KB
```

```
[9]: def analyze_column_data(df, column_name):
    """
    Verilen sütunun eksik, düzgün, uyumsuz veri sayısını ve kategorik
    değerlerini yazdırır.

    :param df: Pandas DataFrame
    :param column_name: İncelenecek sütun adı
    """
    if column_name in df.columns:
        dtype = df[column_name].dtype

        # Sadece kategorik türdeki sütunlar için işlem yap
        if dtype.name == 'category':
            missing_count = df[column_name].isnull().sum()
            total_count = df.shape[0]
            valid_count = total_count - missing_count

            print(f"Sütun: {column_name}")
```

```

print(f" Veri Türü: {dtype}")
print(f" Eksik Veri Sayısı: {missing_count}")
print(f" Düzgün Veri Sayısı: {valid_count}")

valid_categories = df[column_name].cat.categories
invalid_values = df[~df[column_name].isin(valid_categories) &
~df[column_name].isna()]
invalid_count = invalid_values.shape[0]

print(f" Uyumsuz Veri Sayısı: {invalid_count}")
print(f" Kategoriler: {valid_categories}")
print(df[column_name].value_counts())

print("-" * 70)
else:
    print(f"{column_name} sütunu kategorik türde değil.")
else:
    print(f"{column_name} sütunu veri çerçevesinde bulunamadı.")

```

```
[10]: analyze_column_data(df, 'Cinsiyet')
```

```

Sütun: Cinsiyet
Veri Türü: category
Eksik Veri Sayısı: 778
Düzgün Veri Sayısı: 1579
Uyumsuz Veri Sayısı: 0
Kategoriler: Index(['Female', 'Male'], dtype='object')
Cinsiyet
Female      872
Male        707
Name: count, dtype: int64
-----

```

```
[11]: analyze_column_data(df, 'Uyruk')
```

```

Sütun: Uyruk
Veri Türü: category
Eksik Veri Sayısı: 0
Düzgün Veri Sayısı: 2357
Uyumsuz Veri Sayısı: 0
Kategoriler: Index(['Turkiye'], dtype='object')
Uyruk
Turkiye     2357
Name: count, dtype: int64
-----

```

```
[12]: analyze_column_data(df, 'I1')
```



```
Sütun: İl
Veri Türü: category
Eksik Veri Sayısı: 227
Düzenli Veri Sayısı: 2130
Uyumsuz Veri Sayısı: 0
Kategoriler: Index(['Adana', 'Ankara', 'Antalya', 'Bursa', 'Canakkale',
'Eskisehir',
'Istanbul', 'Izmir', 'Kayseri', 'Malatya', 'Mersin', 'Samsun',
'Trabzon'],
dtype='object')
```

```
İl
Adana      220
Mersin     207
Trabzon    195
Ankara     179
Canakkale  179
Eskisehir  174
Antalya    171
Bursa      169
Samsun     153
Istanbul   144
Izmir      121
Kayseri    109
Malatya    109
Name: count, dtype: int64
```

```
[13]: analyze_column_data(df, 'İlac_Adi')
```

```
Sütun: İlac_Adi
Veri Türü: category
Eksik Veri Sayısı: 0
Düzenli Veri Sayısı: 2357
Uyumsuz Veri Sayısı: 0
Kategoriler: Index(['Levomilnacipran', 'Mometasone furoate', 'Prednicarbate
cream',
'alprazolam', 'alprazolam concentrate', 'alprazolam er',
'alprazolam intensol', 'alprazolam odt', 'amitriptyline hcl',
'amoksapin',
...
'valproic acid solution', 'venlafaxine', 'venlafaxine xr', 'vilazodone',
'vortioxetine', 'warfarin sodium', 'zaleplon', 'ziprasidone',
'ziprasidone vial', 'zolpidem tablet, sublingual'],
dtype='object', length=151)

İlac_Adi
chlordiazepoxide-amitriptyline      29
hydrocortisone cream                 27
lurasidone                          27
```

```

desvenlafaxine succinate tablet, extended release 24 hr    26
loxapine                                                    26
..
citalopram                                                  8
risperdal consta syringe                                    8
pethidine hcl                                               8
desoximetasone ointment                                     8
lithium carbonate                                           7
Name: count, Length: 151, dtype: int64
-----

```

```
[14]: analyze_column_data(df, 'Yan_Etki')
```

```

Sütun: Yan_Etki
Veri Türü: category
Eksik Veri Sayısı: 0
Düzgün Veri Sayısı: 2357
Uyumsuz Veri Sayısı: 0
Kategoriler: Index(['Agizda Farkli Bir Tat', 'Az Uyuma', 'Bas Agrisi',
'Bulanti',
'Carpinti', 'Deride Morarma', 'Gec Bosalma', 'Gormede Bulaniklik',
'Gucsuzluk', 'Huzursuzluk', 'Ishal', 'Istah Artisi', 'Kabizlik',
'Karin Agrisi', 'Kas Agrisi', 'Mide Bulantisi', 'Sinirlilik',
'Tansiyon Dusuklugu', 'Tansiyon Yukselme', 'Terleme', 'Uykululuk Hali',
'Yorgunluk'],
dtype='object')
Yan_Etki
Agizda Farkli Bir Tat    229
Tansiyon Yukselme        227
Yorgunluk                204
Gormede Bulaniklik       170
Kabizlik                 156
Ishal                   145
Az Uyuma                 141
Terleme                  97
Gec Bosalma              81
Gucsuzluk                81
Sinirlilik               81
Mide Bulantisi           80
Tansiyon Dusuklugu       74
Carpinti                 72
Bas Agrisi               71
Istah Artisi             71
Karin Agrisi             69
Uykululuk Hali          68
Bulanti                  64
Huzursuzluk             63
Kas Agrisi              58

```

Deride Morarma 55
Name: count, dtype: int64

```
[15]: analyze_column_data(df, 'Alerjilerim')  
# Alerji sınıfında boş değerlere "YOK" atanmalı. İnsanların alerjisi olmayabilir.
```

```
Sütun: Alerjilerim  
Veri Türü: category  
Eksik Veri Sayısı: 484  
Düzgün Veri Sayısı: 1873  
Uyumsuz Veri Sayısı: 0  
Kategoriler: Index(['Bal', 'Balik', 'Bugday', 'Ceviz', 'Cikolata', 'Deniz  
Urunler',  
                    'Domates', 'Elma', 'Havyar', 'Ispanak', 'Ispanakgiller', 'Istiridye',  
                    'Kabak Cekirdegi', 'Kahve', 'Kakao', 'Karaciger', 'Kedi', 'Kefir',  
                    'Kolali Icecek', 'Kopek', 'Laktoz', 'Lateks', 'Muz', 'Pancar',  
                    'Sari Kantaron', 'Sut', 'Toz', 'Yogurt'],  
                    dtype='object')
```

Alerjilerim	
Domates	118
Karaciger	109
Pancar	100
Balik	98
Havyar	97
Ispanakgiller	95
Kopek	91
Kefir	91
Yogurt	89
Deniz Urunler	87
Muz	81
Kabak Cekirdegi	77
Kahve	76
Sari Kantaron	75
Toz	70
Laktoz	60
Bugday	55
Elma	53
Cikolata	53
Ispanak	48
Istiridye	46
Bal	43
Kedi	38
Lateks	32
Ceviz	30
Sut	27
Kakao	22
Kolali Icecek	12

Name: count, dtype: int64

```
[16]: analyze_column_data(df, 'Kan Grubu')
```

Sütun: Kan Grubu

Veri Türü: category

Eksik Veri Sayısı: 347

Düzgün Veri Sayısı: 2010

Uyumsuz Veri Sayısı: 0

Kategoriler: Index(['O RH+', 'O RH-', 'A RH+', 'A RH-', 'AB RH+', 'AB RH-', 'B RH+',
'B RH-'],
dtype='object')

Kan Grubu

AB RH- 421

O RH+ 275

B RH+ 256

AB RH+ 250

B RH- 233

O RH- 232

A RH+ 198

A RH- 145

Name: count, dtype: int64

```
[17]: analyze_column_data(df, 'Kronik Hastaliklarim')  
# Kronik Hastaliklarim sınıfında boş değerlere "YOK" atanmalı.İnsanların Kronik  
↪Hastalığı olmayabilir.
```

Sütun: Kronik Hastaliklarim

Veri Türü: category

Eksik Veri Sayısı: 392

Düzgün Veri Sayısı: 1965

Uyumsuz Veri Sayısı: 0

Kategoriler: Index(['Alzheimer', 'Alzheimer, Astim', 'Alzheimer, Diger',
'Alzheimer, Guatr',
'Alzheimer, Hipertansiyon', 'Astim', 'Astim, Diger', 'Astim, Diyabet',
'Astim, Guatr', 'Astim, Hipertansiyon', 'Astim, Kalp Hastaliklari',
'Astim, Kan Hastaliklari', 'Astim, Kemik Erimesi', 'Diyabet, Alzheimer',
'Diyabet, Astim', 'Diyabet, Diger', 'Diyabet, Guatr',
'Diyabet, Hipertansiyon', 'Diyabet, KOAH', 'Diyabet, Kalp Hastaliklari',
'Guatr', 'Guatr, Diger', 'Guatr, Diyabet', 'Guatr, Hipertansiyon',
'Guatr, KOAH', 'Guatr, Kan Hastaliklari', 'Guatr, Kanser',
'Hipertansiyon', 'Hipertansiyon, Alzheimer', 'Hipertansiyon, Astim',
'Hipertansiyon, Diger', 'Hipertansiyon, Diyabet', 'Hipertansiyon, KOAH',
'Hipertansiyon, Kalp Hastaliklari', 'Hipertansiyon, Kan Hastaliklari',
'Hipertansiyon, Kanser', 'KOAH', 'KOAH, Alzheimer', 'KOAH, Diger',

```

'KOAHA, Diyabet', 'KOAHA, Guatr', 'KOAHA, Hipertansiyon',
'KOAHA, Kalp Hastalıkları', 'KOAHA, Kan Hastalıkları', 'KOAHA, Kanser',
'KOAHA, Kemik Erimesi', 'Kalp Hastalıkları',
'Kalp Hastalıkları, Alzheimer', 'Kalp Hastalıkları, Astim',
'Kalp Hastalıkları, Diyabet', 'Kalp Hastalıkları, Guatr',
'Kalp Hastalıkları, KOAH', 'Kalp Hastalıkları, Kan Hastalıkları',
'Kalp Hastalıkları, Kanser', 'Kan Hastalıkları, Alzheimer',
'Kan Hastalıkları, Astim', 'Kan Hastalıkları, Diger',
'Kan Hastalıkları, Diyabet', 'Kan Hastalıkları, Hipertansiyon',
'Kan Hastalıkları, KOAH', 'Kan Hastalıkları, Kalp Hastalıkları',
'Kan Hastalıkları, Kanser', 'Kanser', 'Kanser, Alzheimer',
'Kanser, Astim', 'Kanser, Diger', 'Kanser, Diyabet', 'Kanser, Guatr',
'Kanser, Hipertansiyon', 'Kanser, Kalp Hastalıkları',
'Kanser, Kan Hastalıkları', 'Kanser, Kemik Erimesi', 'Kemik Erimesi',
'Kemik Erimesi, Alzheimer', 'Kemik Erimesi, Diger',
'Kemik Erimesi, Diyabet', 'Kemik Erimesi, Hipertansiyon',
'Kemik Erimesi, KOAH', 'Kemik Erimesi, Kalp Hastalıkları',
'Kemik Erimesi, Kan Hastalıkları'],
dtype='object')
Kronik Hastalıklarım
Alzheimer, Diger 61
Kan Hastalıkları, Hipertansiyon 58
Alzheimer, Astim 57
KOAHA, Guatr 55
Kemik Erimesi, Kan Hastalıkları 48
..
Astim, Guatr 8
Hipertansiyon, Alzheimer 7
Astim, Kan Hastalıkları 7
Kanser, Kan Hastalıkları 6
Astim 3
Name: count, Length: 80, dtype: int64
-----

```

```

[18]: analyze_column_data(df, 'Baba Kronik Hastalıkları')
# Baba Kronik Hastalıkları sınıfında boş değerlere "YOK" atanmalı.İnsanların
↳Kronik Hastalığı olmayabilir.

```

```

Sütun: Baba Kronik Hastalıkları
Veri Türü: category
Eksik Veri Sayısı: 156
Düzgün Veri Sayısı: 2201
Uyumsuz Veri Sayısı: 0
Kategoriler: Index([' ', 'Alzheimer', 'Alzheimer, Astim', 'Alzheimer, Diger',
'Alzheimer, Diyabet', 'Alzheimer, Guatr', 'Alzheimer, Hipertansiyon',
'Alzheimer, Kalp Hastalıkları', 'Alzheimer, Kemik Erimesi', 'Astim',
'Astim, Diger', 'Astim, Diyabet', 'Astim, Guatr',
'Astim, Hipertansiyon', 'Astim, KOAH', 'Astim, Kalp Hastalıkları',

```

```

'Astim, Kanser', 'Astim, Kemik Erimesi', 'Diger', 'Diyabet',
'Diyabet, Alzheimer', 'Diyabet, Diger', 'Diyabet, Diyabet',
'Diyabet, Guatr', 'Diyabet, Hipertansiyon', 'Diyabet, KOAH',
'Diyabet, Kalp Hastalıkları', 'Diyabet, Kan Hastalıkları',
'Diyabet, Kemik Erimesi', 'Guatr', 'Guatr, Alzheimer', 'Guatr, Diger',
'Guatr, Hipertansiyon', 'Guatr, KOAH', 'Guatr, Kalp Hastalıkları',
'Guatr, Kan Hastalıkları', 'Guatr, Kemik Erimesi', 'Hipertansiyon',
'Hipertansiyon, Alzheimer', 'Hipertansiyon, Astim',
'Hipertansiyon, Diger', 'Hipertansiyon, Diyabet', 'Hipertansiyon, KOAH',
'Hipertansiyon, Kan Hastalıkları', 'Hipertansiyon, Kanser',
'Hipertansiyon, Kemik Erimesi', 'KOAH', 'KOAH, Alzheimer',
'KOAH, Astim', 'KOAH, Diger', 'KOAH, Diyabet', 'KOAH, Guatr',
'KOAH, Kalp Hastalıkları', 'KOAH, Kan Hastalıkları', 'KOAH, Kanser',
'KOAH, Kemik Erimesi', 'Kalp Hastalıkları', 'Kalp Hastalıkları, Astim',
'Kalp Hastalıkları, Diger', 'Kalp Hastalıkları, Guatr',
'Kalp Hastalıkları, Hipertansiyon', 'Kalp Hastalıkları, KOAH',
'Kalp Hastalıkları, Kan Hastalıkları', 'Kalp Hastalıkları, Kanser',
'Kalp Hastalıkları, Kemik Erimesi', 'Kan Hastalıkları',
'Kan Hastalıkları, Astim', 'Kan Hastalıkları, Diger',
'Kan Hastalıkları, Hipertansiyon', 'Kan Hastalıkları, KOAH',
'Kan Hastalıkları, Kanser', 'Kan Hastalıkları, Kemik Erimesi', 'Kanser',
'Kanser, Alzheimer', 'Kanser, Astim', 'Kanser, Diger',
'Kanser, Diyabet', 'Kanser, Guatr', 'Kanser, Hipertansiyon',
'Kanser, KOAH', 'Kanser, Kalp Hastalıkları', 'Kanser, Kan Hastalıkları',
'Kanser, Kemik Erimesi', 'Kemik Erimesi', 'Kemik Erimesi, Alzheimer',
'Kemik Erimesi, Diger', 'Kemik Erimesi, Diyabet',
'Kemik Erimesi, Guatr', 'Kemik Erimesi, Hipertansiyon',
'Kemik Erimesi, KOAH', 'Kemik Erimesi, Kan Hastalıkları',
'Kemik Erimesi, Kanser'],
dtype='object')
Baba Kronik Hastalıkları
Alzheimer, Diyabet          64
Kanser                      62
                             55
Alzheimer                   53
Kalp Hastalıkları, Kan Hastalıkları  51
                             ..
Alzheimer, Kalp Hastalıkları    9
Astim, KOAH                   9
KOAH, Astim                   9
Astim                         8
KOAH, Alzheimer               6
Name: count, Length: 92, dtype: int64
-----

```

```
[19]: analyze_column_data(df, 'Anne Kronik Hastalıkları')
```

Anne Kronik Hastalıkları sınıfında boş değerlere "YOK" atanmalı. İnsanların
↪Kronik Hastalığı olmayabilir.

Sütun: Anne Kronik Hastalıkları

Veri Türü: category

Eksik Veri Sayısı: 217

Düzgün Veri Sayısı: 2140

Uyumsuz Veri Sayısı: 0

Kategoriler: Index([' ', 'Alzheimer', 'Alzheimer, Astim', 'Alzheimer, Diger',
'Alzheimer, Hipertansiyon', 'Alzheimer, KOAH',
'Alzheimer, Kalp Hastalıkları', 'Astim', 'Astim, Diger',
'Astim, Diyabet', 'Astim, Guatr', 'Astim, Hipertansiyon', 'Astim, KOAH',
'Astim, Kalp Hastalıkları', 'Astim, Kan Hastalıkları',
'Astim, Kemik Erimesi', 'Diger', 'Diyabet', 'Diyabet, Astim',
'Diyabet, Diger', 'Diyabet, Diyabet', 'Diyabet, Guatr',
'Diyabet, Hipertansiyon', 'Diyabet, KOAH', 'Diyabet, Kalp Hastalıkları',
'Diyabet, Kan Hastalıkları', 'Diyabet, Kemik Erimesi', 'Guatr',
'Guatr, Alzheimer', 'Guatr, Astim', 'Guatr, Diger', 'Guatr, Diyabet',
'Guatr, Hipertansiyon', 'Guatr, Kan Hastalıkları',
'Guatr, Kemik Erimesi', 'Hipertansiyon', 'Hipertansiyon, Alzheimer',
'Hipertansiyon, Diyabet', 'Hipertansiyon, Guatr',
'Hipertansiyon, Kalp Hastalıkları', 'Hipertansiyon, Kan Hastalıkları',
'Hipertansiyon, Kanser', 'Hipertansiyon, Kemik Erimesi', 'KOAH',
'KOAH, Alzheimer', 'KOAH, Astim', 'KOAH, Diger', 'KOAH, Guatr',
'KOAH, Hipertansiyon', 'KOAH, Kalp Hastalıkları', 'KOAH, Kemik Erimesi',
'Kalp Hastalıkları', 'Kalp Hastalıkları, Alzheimer',
'Kalp Hastalıkları, Astim', 'Kalp Hastalıkları, Diger',
'Kalp Hastalıkları, Diyabet', 'Kalp Hastalıkları, Guatr',
'Kalp Hastalıkları, Hipertansiyon', 'Kalp Hastalıkları, KOAH',
'Kalp Hastalıkları, Kemik Erimesi', 'Kan Hastalıkları',
'Kan Hastalıkları, Alzheimer', 'Kan Hastalıkları, Astim',
'Kan Hastalıkları, Diger', 'Kan Hastalıkları, Diyabet',
'Kan Hastalıkları, Hipertansiyon', 'Kan Hastalıkları, Kanser',
'Kan Hastalıkları, Kemik Erimesi', 'Kanser', 'Kanser, Alzheimer',
'Kanser, Diger', 'Kanser, Diyabet', 'Kanser, Guatr',
'Kanser, Hipertansiyon', 'Kanser, KOAH', 'Kanser, Kalp Hastalıkları',
'Kanser, Kemik Erimesi', 'Kemik Erimesi', 'Kemik Erimesi, Alzheimer',
'Kemik Erimesi, Astim', 'Kemik Erimesi, Diger',
'Kemik Erimesi, Diyabet', 'Kemik Erimesi, Kan Hastalıkları',
'Kemik Erimesi, Kanser'],
dtype='object')

Anne Kronik Hastalıkları

Alzheimer, Astim	110
KOAH	70
	65
Kanser, Guatr	61
Kemik Erimesi, Kan Hastalıkları	58

```

...
Astim, Diyabet      8
KOAHA, Kemik Erimesi 8
Kan Hastalıkları, Hipertansiyon 8
Diyabet, Kalp Hastalıkları 6
Kemik Erimesi, Kanser 3
Name: count, Length: 84, dtype: int64
-----

```

```

[20]: analyze_column_data(df, 'Kiz Kardes Kronik Hastalıkları')
# Kiz Kardes Kronik Hastalıkları sınıfında boş değerlere "YOK" atanmalı.
↳ İnsanların Kronik Hastalığı olmayabilir.

```

Sütun: Kiz Kardes Kronik Hastalıkları

Veri Türü: category

Eksik Veri Sayısı: 97

Düzenli Veri Sayısı: 2260

Uyumsuz Veri Sayısı: 0

Kategoriler: Index([' ', 'Alzheimer', 'Alzheimer, Astim', 'Alzheimer, Diger', 'Alzheimer, Diyabet', 'Alzheimer, Guatr', 'Alzheimer, Hipertansiyon', 'Alzheimer, Kanser', 'Alzheimer, Kemik Erimesi', 'Astim', 'Astim, Diger', 'Astim, Diyabet', 'Astim, Guatr', 'Astim, Hipertansiyon', 'Astim, Kan Hastalıkları', 'Astim, Kanser', 'Astim, Kemik Erimesi', 'Diger', 'Diyabet, Alzheimer', 'Diyabet, Astim', 'Diyabet, Diger', 'Diyabet, KOAH', 'Diyabet, Kan Hastalıkları', 'Diyabet, Kanser', 'Diyabet, Kemik Erimesi', 'Guatr', 'Guatr, Astim', 'Guatr, Diger', 'Guatr, Diyabet', 'Guatr, Hipertansiyon', 'Guatr, KOAH', 'Guatr, Kan Hastalıkları', 'Guatr, Kemik Erimesi', 'Hipertansiyon', 'Hipertansiyon, Alzheimer', 'Hipertansiyon, Astim', 'Hipertansiyon, Diyabet', 'Hipertansiyon, Guatr', 'Hipertansiyon, Kalp Hastalıkları', 'KOAH', 'KOAH, Alzheimer', 'KOAH, Astim', 'KOAH, Diyabet', 'KOAH, Guatr', 'KOAH, Kan Hastalıkları', 'KOAH, Kemik Erimesi', 'Kalp Hastalıkları', 'Kalp Hastalıkları, Alzheimer', 'Kalp Hastalıkları, Astim', 'Kalp Hastalıkları, Diger', 'Kalp Hastalıkları, Hipertansiyon', 'Kalp Hastalıkları, KOAH', 'Kalp Hastalıkları, Kan Hastalıkları', 'Kalp Hastalıkları, Kanser', 'Kalp Hastalıkları, Kemik Erimesi', 'Kan Hastalıkları', 'Kan Hastalıkları, Alzheimer', 'Kan Hastalıkları, Astim', 'Kan Hastalıkları, Diger', 'Kan Hastalıkları, Diyabet', 'Kan Hastalıkları, Hipertansiyon', 'Kan Hastalıkları, KOAH', 'Kan Hastalıkları, Kalp Hastalıkları', 'Kan Hastalıkları, Kanser', 'Kanser', 'Kanser, Alzheimer', 'Kanser, Astim', 'Kanser, Diger', 'Kanser, Diyabet', 'Kanser, Guatr', 'Kanser, Hipertansiyon', 'Kanser, KOAH', 'Kanser, Kalp Hastalıkları', 'Kanser, Kan Hastalıkları', 'Kemik Erimesi', 'Kemik Erimesi, Alzheimer', 'Kemik Erimesi, Astim', 'Kemik Erimesi, Diger', 'Kemik Erimesi, Diyabet', 'Kemik Erimesi, Guatr', 'Kemik Erimesi, Hipertansiyon', 'Kemik Erimesi, KOAH',


```

        'Kemik Erimesi, Kalp Hastaliklari', 'Kemik Erimesi, Kan Hastaliklari',
        'Kemik Erimesi, Kanser'],
        dtype='object')
Kiz Kardes Kronik Hastaliklari
                                     168
Kanser, Hipertansiyon                70
Alzheimer, Astim                     61
Kanser                               61
Guatr                                59
...
Diyabet, Astim                       9
Hipertansiyon, Diyabet               8
Kemik Erimesi, Kanser               8
Astim, Kanser                       6
Hipertansiyon, Kalp Hastaliklari     6
Name: count, Length: 85, dtype: int64
-----

```

```

[21]: analyze_column_data(df, 'Erkek Kardes Kronik Hastaliklari')
      # Erkek Kardes Kronik Hastaliklari sınıfında boş değerlere "YOK" atanmalı.
      ↪ İnsanların Kronik Hastalığı olmayabilir.

```

```

Sütun: Erkek Kardes Kronik Hastaliklari
Veri Türü: category
Eksik Veri Sayısı: 121
Düzgün Veri Sayısı: 2236
Uyumsuz Veri Sayısı: 0
Kategoriler: Index([' ', 'Alzheimer', 'Alzheimer, Astim', 'Alzheimer, Guatr',
                    'Alzheimer, Hipertansiyon', 'Alzheimer, KOAH',
                    'Alzheimer, Kalp Hastaliklari', 'Alzheimer, Kanser',
                    'Alzheimer, Kemik Erimesi', 'Astim', 'Astim, Diyabet', 'Astim, Guatr',
                    'Astim, Hipertansiyon', 'Astim, KOAH', 'Astim, Kan Hastaliklari',
                    'Astim, Kanser', 'Astim, Kemik Erimesi', 'Diger', 'Diyabet',
                    'Diyabet, Alzheimer', 'Diyabet, Astim', 'Diyabet, Diger',
                    'Diyabet, Diyabet', 'Diyabet, KOAH', 'Diyabet, Kalp Hastaliklari',
                    'Diyabet, Kan Hastaliklari', 'Diyabet, Kanser',
                    'Diyabet, Kemik Erimesi', 'Guatr', 'Guatr, Astim', 'Guatr, Diger',
                    'Guatr, Diyabet', 'Guatr, Hipertansiyon', 'Guatr, KOAH',
                    'Guatr, Kalp Hastaliklari', 'Guatr, Kan Hastaliklari', 'Guatr, Kanser',
                    'Guatr, Kemik Erimesi', 'Hipertansiyon', 'Hipertansiyon, Alzheimer',
                    'Hipertansiyon, Astim', 'Hipertansiyon, Diger',
                    'Hipertansiyon, Diyabet', 'Hipertansiyon, Guatr', 'Hipertansiyon, KOAH',
                    'Hipertansiyon, Kalp Hastaliklari', 'Hipertansiyon, Kan Hastaliklari',
                    'Hipertansiyon, Kemik Erimesi', 'KOAH', 'KOAH, Alzheimer',
                    'KOAH, Diger', 'KOAH, Diyabet', 'KOAH, Kalp Hastaliklari',
                    'KOAH, Kan Hastaliklari', 'Kalp Hastaliklari',
                    'Kalp Hastaliklari, Alzheimer', 'Kalp Hastaliklari, Astim',
                    'Kalp Hastaliklari, Diger', 'Kalp Hastaliklari, Diyabet',

```

```

'Kalp Hastaliklari, Guatr', 'Kalp Hastaliklari, Hipertansiyon',
'Kalp Hastaliklari, KOAH', 'Kalp Hastaliklari, Kan Hastaliklari',
'Kalp Hastaliklari, Kanser', 'Kalp Hastaliklari, Kemik Erimesi',
'Kan Hastaliklari', 'Kan Hastaliklari, Alzheimer',
'Kan Hastaliklari, Diyabet', 'Kan Hastaliklari, Hipertansiyon',
'Kan Hastaliklari, KOAH', 'Kan Hastaliklari, Kanser',
'Kan Hastaliklari, Kemik Erimesi', 'Kanser', 'Kanser, Alzheimer',
'Kanser, Diger', 'Kanser, Diyabet', 'Kanser, Guatr', 'Kanser, KOAH',
'Kanser, Kalp Hastaliklari', 'Kanser, Kan Hastaliklari',
'Kanser, Kemik Erimesi', 'Kemik Erimesi', 'Kemik Erimesi, Alzheimer',
'Kemik Erimesi, Diger', 'Kemik Erimesi, Diyabet',
'Kemik Erimesi, Guatr', 'Kemik Erimesi, Hipertansiyon',
'Kemik Erimesi, KOAH', 'Kemik Erimesi, Kalp Hastaliklari',
'Kemik Erimesi, Kan Hastaliklari'],
dtype='object')
Erkek Kardes Kronik Hastaliklari
136
Kemik Erimesi
71
Diger
68
Alzheimer, Astim
51
Alzheimer
49
...
Kalp Hastaliklari, Kanser
9
Kanser, Diyabet
9
Hipertansiyon, KOAH
8
Guatr
8
Hipertansiyon, Kan Hastaliklari
6
Name: count, Length: 90, dtype: int64
-----

```

Yukarıdaki çıktılarından anladığımız kadarıyla:

1. Bazı sütunlarda boş olması gereken değerleri NAN atandığı için boş değer olarak sayıp diğer verilere göre doldurmaya çalışmamak için inceleyip karar verilen satırlarda boş veya NAN olan değerlere “YOK” değeri atanacaktır.

2. Bazı sütunlarda kategorik değişken incelemesi sırasında “,” ile birden fazla unique değerın yazılmasından dolayı sanki çok fazla kategori varmış gibi olduğu farkedilmiş olup aşağıda buna özel bir method ile gerçek sayısı öğrenilecek ve modelin anlayabilmesi amacıyla tüm bu kategoriler yeni bir sütun adı olarak kaydedilip 0,1 değerleri ile satırlara atanacaktır. Bu sorunlu sütunlar silinecektir.

```

[22]: def replace_nan_with_yok(df, column_name):
      """
      Verilen sütundaki NaN ve boş değerleri 'YOK' ile değiştirir.

      :param df: Pandas DataFrame
      :param column_name: Güncellenecek sütun adı

```

```

"""
if column_name in df.columns:
    # Kategorik sütun ise 'YOK' kategorisini ekle
    if df[column_name].dtype.name == 'category':
        if 'YOK' not in df[column_name].cat.categories:
            df[column_name] = df[column_name].cat.add_categories(['YOK'])

    # NaN değerlerini 'YOK' ile değiştir
    df[column_name] = df[column_name].fillna('YOK')

    # Boş ve sadece boşluklardan oluşan değerleri 'YOK' ile değiştir
    df[column_name] = df[column_name].replace(r'^\s*$', 'YOK', regex=True)

    print(f"{column_name} sütunundaki NaN ve boş değerler 'YOK' ile
↪değiştirildi.")
else:
    print(f"{column_name} sütunu veri çerçevesinde bulunamadı.")

```

[23]: `replace_nan_with_yok(df, 'Kronik Hastaliklarim')`

Kronik Hastaliklarim sütunundaki NaN ve boş değerler 'YOK' ile değiştirildi.

[24]: `replace_nan_with_yok(df, 'Baba Kronik Hastaliklari')`

Baba Kronik Hastaliklari sütunundaki NaN ve boş değerler 'YOK' ile değiştirildi.

[25]: `replace_nan_with_yok(df, 'Anne Kronik Hastaliklari')`

Anne Kronik Hastaliklari sütunundaki NaN ve boş değerler 'YOK' ile değiştirildi.

[26]: `replace_nan_with_yok(df, 'Kiz Kardes Kronik Hastaliklari')`

Kiz Kardes Kronik Hastaliklari sütunundaki NaN ve boş değerler 'YOK' ile değiştirildi.

[27]: `replace_nan_with_yok(df, 'Erkek Kardes Kronik Hastaliklari')`

Erkek Kardes Kronik Hastaliklari sütunundaki NaN ve boş değerler 'YOK' ile değiştirildi.

[28]: `def split_analyze_column_data(df, column_name):`

```

"""
    Verilen sütunun değerlerini split ederek her bir parçayı ayrı kategori
↪olarak değerlendirir
    ve benzersiz kategorilerin sayılarını yazdırır.

:param df: Pandas DataFrame
:param column_name: İncelenecek sütun adı

```

```

"""
if column_name in df.columns:
    dtype = df[column_name].dtype

    # Sadece kategorik türdeki sütunlar için işlem yap
    if dtype.name == 'category':
        # Split işlemi
        split_values = df[column_name].dropna().str.split(',', expand=True)
        new_values = pd.unique(split_values.values.ravel())
        new_values = new_values[~pd.isnull(new_values)] # NaN değerleri

        ↪ çıkar

        # Her bir benzersiz kategori için sayıyı hesapla
        category_counts = {category: 0 for category in new_values}
        for index, row in df.iterrows():
            if pd.notnull(row[column_name]):
                for value in row[column_name].split(','):
                    value = value.strip()
                    if value in category_counts:
                        category_counts[value] += 1

        # Kategorileri alfabetik sıraya göre sırala
        sorted_categories = sorted(category_counts.keys())

        # Split sonrası eksik ve düzgün veri sayısını hesapla
        missing_count = df[column_name].isnull().sum()
        valid_count = sum(1 for row in split_values.values.flatten() if pd.
        ↪ notnull(row))

        # Sonuçları yazdır
        print(f"Sütun: {column_name}")
        print(f"Veri Türü: {dtype}")
        print(f"Eksik Veri Sayısı: {missing_count}")
        print(f"Düğüün Veri Sayısı: {valid_count}")
        print(f"Kategoriler: {sorted_categories}")

        print("Kategorilerin Sayıları:")
        max_length = max(len(category) for category in sorted_categories)
        ↪ # Kategori isimlerinin en uzun uzunluğu
        for category in sorted_categories:
            count = category_counts[category]
            print(f"{category:<{max_length}}: {count}")

        print("-" * 70)
    else:
        print(f"{column_name} sütunu kategorik türde değıil.")
else:

```

```
print(f"{column_name} sütunu veri çerçevesinde bulunamadı.")
```

```
[29]: split_analyze_column_data(df, 'Kronik Hastaliklarim')  
# Kronik Hastaliklarim sınıfında boş değerlere "YOK" atanmalı.İnsanların Kronik  
↪Hastalığı olmayabilir.
```

Sütun: Kronik Hastaliklarim

Veri Türü: category

Eksik Veri Sayısı: 0

Düzenli Veri Sayısı: 4136

Kategoriler: ['Alzheimer', 'Astim', 'Diger', 'Diyabet', 'Guatr',
'Hipertansiyon', 'KOAHA', 'Kalp Hastaliklari', 'Kan Hastaliklari', 'Kanser',
'Kemik Erimesi', 'YOK']

Kategorilerin Sayıları:

Alzheimer	: 343
Astim	: 296
Diger	: 248
Diyabet	: 396
Guatr	: 305
Hipertansiyon	: 461
KOAHA	: 421
Kalp Hastaliklari	: 327
Kan Hastaliklari	: 383
Kanser	: 273
Kemik Erimesi	: 291
YOK	: 392

```
[30]: split_analyze_column_data(df, 'Baba Kronik Hastaliklari')  
# Baba Kronik Hastaliklari sınıfında boş değerlere "YOK" atanmalı.İnsanların  
↪Kronik Hastalığı olmayabilir.
```

Sütun: Baba Kronik Hastaliklari

Veri Türü: category

Eksik Veri Sayısı: 0

Düzenli Veri Sayısı: 4122

Kategoriler: ['Alzheimer', 'Astim', 'Diger', 'Diyabet', 'Guatr',
'Hipertansiyon', 'KOAHA', 'Kalp Hastaliklari', 'Kan Hastaliklari', 'Kanser',
'Kemik Erimesi', 'YOK']

Kategorilerin Sayıları:

Alzheimer	: 333
Astim	: 310
Diger	: 230
Diyabet	: 323
Guatr	: 331
Hipertansiyon	: 408
KOAHA	: 365

```
Kalp Hastaliklari: 358
Kan Hastaliklari : 377
Kanser           : 437
Kemik Erimesi    : 439
YOK              : 211
```

```
[31]: split_analyze_column_data(df, 'Anne Kronik Hastaliklari')
# Anne Kronik Hastaliklari sınıfında boş değerlere "YOK" atanmalı.İnsanların
↳Kronik Hastalığı olmayabilir.
```

Sütun: Anne Kronik Hastaliklari

```
Veri Türü: category
Eksik Veri Sayısı: 0
Düzgün Veri Sayısı: 4074
Kategoriler: ['Alzheimer', 'Astim', 'Diger', 'Diyabet', 'Guatr',
'Hipertansiyon', 'KOAHA', 'Kalp Hastaliklari', 'Kan Hastaliklari', 'Kanser',
'Kemik Erimesi', 'YOK']
Kategorilerin Sayıları:
Alzheimer      : 413
Astim          : 472
Diger          : 204
Diyabet        : 269
Guatr          : 344
Hipertansiyon  : 355
KOAHA          : 321
Kalp Hastaliklari: 473
Kan Hastaliklari : 302
Kanser         : 261
Kemik Erimesi  : 378
YOK            : 282
```

```
[32]: split_analyze_column_data(df, 'Kiz Kardes Kronik Hastaliklari')
# Kiz Kardes Kronik Hastaliklari sınıfında boş değerlere "YOK" atanmalı.
↳İnsanların Kronik Hastalığı olmayabilir.
```

Sütun: Kiz Kardes Kronik Hastaliklari

```
Veri Türü: category
Eksik Veri Sayısı: 0
Düzgün Veri Sayısı: 4052
Kategoriler: ['Alzheimer', 'Astim', 'Diger', 'Diyabet', 'Guatr',
'Hipertansiyon', 'KOAHA', 'Kalp Hastaliklari', 'Kan Hastaliklari', 'Kanser',
'Kemik Erimesi', 'YOK']
Kategorilerin Sayıları:
Alzheimer      : 444
Astim          : 421
Diger          : 173
```

```
Diyabet          : 379
Guatr            : 323
Hipertansiyon    : 322
KOAHA           : 314
Kalp Hastaliklari: 227
Kan Hastaliklari : 402
Kanser          : 351
Kemik Erimesi    : 431
YOK             : 265
```

```
[33]: split_analyze_column_data(df, 'Erkek Kardes Kronik Hastaliklari')
# Erkek Kardes Kronik Hastaliklari sinifinda bos degerlere "YOK" atanmalir.
↳ İnsanların Kronik Hastalığı olmayabilir.
```

Sütun: Erkek Kardes Kronik Hastaliklari

Veri Türü: category

Eksik Veri Sayısı: 0

Düzenli Veri Sayısı: 4098

Kategoriler: ['Alzheimer', 'Astim', 'Diger', 'Diyabet', 'Guatr', 'Hipertansiyon', 'KOAHA', 'Kalp Hastaliklari', 'Kan Hastaliklari', 'Kanser', 'Kemik Erimesi', 'YOK']

Kategorilerin Sayıları:

```
Alzheimer        : 391
Astim            : 323
Diger            : 197
Diyabet          : 403
Guatr            : 307
Hipertansiyon    : 233
KOAHA           : 342
Kalp Hastaliklari: 412
Kan Hastaliklari : 389
Kanser          : 380
Kemik Erimesi    : 464
YOK             : 257
```

Aşağıdaki işlem yukarıda 2.numaralı maddede anlattığım işlemin kodlarını içeriyor. satıdaki değerleri onehotencoder mantığıyla sütuna çekip 0,1 ile denetleyerek hem id ve satır düzenini bozmayacağız hem de verilerimiz anlamlı olabilecek.

Bir problem ortaya doğdu ve 5 farklı kişi için aynı hastalıklar var hepsi için yeni sütun eklemek veya her satır'ı birden fazla hastalığa bölmek yerine sunduğum çözüm:

Hastalıklar aynı olduğu için kişilere sayı vereceğim. 0 Yok 1 Sen hasta 2 Baba hasta 3 Sen ve baba hasta 4 Anne hasta 5 Sen ve anne hasta 6 Baba ve anne hasta 7 Sen, baba ve anne hasta 8 Kız kardeş hasta 9 Sen ve kız kardeş hasta 10 Baba ve kız kardeş hasta 11 Sen, baba ve kız

kardeş hasta 12 Anne ve kız kardeş hasta 13 Sen, anne ve kız kardeş hasta 14 Baba, anne ve kız kardeş hasta 15 Sen, baba, anne ve kız kardeş hasta 16 Erkek kardeş hasta 17 Sen ve erkek kardeş hasta 18 Baba ve erkek kardeş hasta 19 Sen, baba ve erkek kardeş hasta 20 Anne ve erkek kardeş hasta 21 Sen, anne ve erkek kardeş hasta 22 Baba, anne ve erkek kardeş hasta 23 Sen, baba, anne ve erkek kardeş hasta 24 Kız kardeş ve erkek kardeş hasta 25 Sen, kız kardeş ve erkek kardeş hasta 26 Baba, kız kardeş ve erkek kardeş hasta 27 Sen, baba, kız kardeş ve erkek kardeş hasta 28 Anne, kız kardeş ve erkek kardeş hasta 29 Sen, anne, kız kardeş ve erkek kardeş hasta 30 Baba, anne, kız kardeş ve erkek kardeş hasta 31 Sen, baba, anne, kız kardeş ve erkek kardeş hasta

```
[34]: import pandas as pd

def kronik_hastalik_analizi(df, column_names):
    unique_categories = set()

    # 1. Split işlemi ile benzersiz kategori isimlerini bul
    for column in column_names:
        if column in df.columns:
            split_values = df[column].dropna().str.split(',', expand=True)
            unique_categories.update(pd.unique(split_values.values.ravel()))

    # None değerleri çıkar
    unique_categories = {cat for cat in unique_categories if cat is not None}

    # 2. Kategorileri alfabetik sıraya koy
    unique_categories = sorted(unique_categories)

    # 3. Yeni sütunlar ekle
    for category in unique_categories:
        if category not in df.columns:
            df[category] = 0

    # 4. Her satır için değerleri kontrol et ve sayıları ata
    def assign_numbers(row, category):
        status = 0 # Başlangıçta "Yok" durumu

        # Kronik hastalıklar, baba, anne, kız kardeş, erkek kardeş için split_
        ↪ yap ve kontrol et
        if category in (row['Kronik Hastalıklarım'] or '').split(', '):
            status += 1 # Sen hasta
        if category in (row['Baba Kronik Hastalıkları'] or '').split(', '):
            status += 2 # Baba hasta
        if category in (row['Anne Kronik Hastalıkları'] or '').split(', '):
            status += 4 # Anne hasta
        if category in (row['Kız Kardeş Kronik Hastalıkları'] or '').split(', '):
            ↪ status += 8 # Kız kardeş hasta
```



```

        if category in (row['Erkek Kardes Kronik Hastaliklari'] or '').split(','):
            status += 16 # Erkek kardeş hasta

        # Durum değerini döndür
        return status

    # Satırları işleyip yeni sütunlara sayıları ata
    for category in unique_categories:
        df[category] = df.apply(lambda row: assign_numbers(row, category),
                                axis=1)

    return df

```

```

[35]: # Fonksiyonu çağırarak ilgili sütunu işleyelim
df = kronik_hastalik_analizi(df, ['Kronik Hastaliklarim', 'Baba Kronik
    Hastaliklari', 'Anne Kronik Hastaliklari', 'Kiz Kardes Kronik Hastaliklari',
    'Erkek Kardes Kronik Hastaliklari'])

# Sonuçları kontrol edelim
df.head(16)

```

```

[35]:
Kullanici_id Cinsiyet Dogum_Tarihi    Uyruk    Il \
0          107    Male  1960-03-01  Turkiye  Canakkale
1          140    Male  1939-10-12  Turkiye   Trabzon
2           2  Female  1976-12-17  Turkiye  Canakkale
3          83    Male  1977-06-17  Turkiye   Adana
4           7  Female  1976-09-03  Turkiye   Izmir
5         131    Male  1982-01-05  Turkiye     NaN
6          61     NaN  1997-01-10  Turkiye   Mersin
7          95     NaN  1997-01-15  Turkiye  Antalya
8           3  Female  1973-08-05  Turkiye     NaN
9          53     NaN  1941-10-16  Turkiye   Adana
10         81     NaN  1955-10-07  Turkiye  Canakkale
11        108     NaN  1992-03-24  Turkiye  Eskisehir
12        121    Male  2001-06-01  Turkiye   Samsun
13         33  Female  1964-05-22  Turkiye   Adana
14        194    Male  1981-03-01  Turkiye  Antalya
15         42    Male  1973-09-09  Turkiye   Ankara

      Ilac_Adi Ilac_Baslangic_Tarihi Ilac_Bitis_Tarihi \
0      trifluoperazine      2022-01-09      2022-03-04
1      fluphenazine hcl      2022-01-09      2022-03-08
2      warfarin sodium      2022-01-11      2022-03-12
3      valproic acid      2022-01-04      2022-03-12
4  carbamazepine extended release      2022-01-13      2022-03-06
5      fluoxetine dr      2022-01-07      2022-03-19

```

6	carbamazepine extended release	2022-01-05	2022-03-07
7	thiothixene	2022-01-10	2022-03-07
8	asenapine tablet, sublingual	2022-01-08	2022-03-17
9	fluphenazine hcl elixir	2022-01-10	2022-03-13
10	quetiapine	2022-01-07	2022-03-13
11	risperdal consta syringe	2022-01-03	2022-03-02
12	mirtazapine	2022-01-12	2022-03-04
13	lamictal xr	2022-01-06	2022-03-13
14	desonide lotion	2022-01-14	2022-03-15
15	olanzapine vial	2022-01-11	2022-03-02

	Yan_Etki	Yan_Etki_Bildirim_Tarihi	...	Diger	Diyabet	Guatr	\
0	Kabizlik	2022-02-19 18:28:43	...	0	0	18	
1	Yorgunluk	2022-02-03 20:48:17	...	2	16	2	
2	Carpinti	2022-02-04 05:29:20	...	16	15	0	
3	Sinirlilik	2022-02-08 01:01:21	...	3	1	0	
4	Agizda Farkli Bir Tat	2022-02-12 05:33:06	...	8	9	0	
5	Agizda Farkli Bir Tat	2022-02-04 08:08:31	...	0	17	0	
6	Agizda Farkli Bir Tat	2022-02-19 17:39:48	...	0	1	0	
7	Deride Morarma	2022-02-09 20:53:54	...	0	3	24	
8	Bas Agrisi	2022-02-19 04:14:26	...	2	0	4	
9	Gormede Bulaniklik	2022-02-09 01:16:50	...	0	0	0	
10	Gucsuzluk	2022-02-01 04:34:33	...	0	3	0	
11	Az Uyuma	2022-02-19 11:09:36	...	0	2	0	
12	Huzursuzluk	2022-02-01 13:27:16	...	0	16	2	
13	Az Uyuma	2022-02-06 14:28:54	...	0	0	0	
14	Mide Bulantisi	2022-02-05 06:50:13	...	0	0	12	
15	Mide Bulantisi	2022-02-05 06:50:13	...	0	0	0	

	Hipertansiyon	KOAH	Kalp Hastaliklari	Kan Hastaliklari	Kanser	\
0	3	4	8	1	0	
1	4	16	4	0	0	
2	0	2	1	0	0	
3	0	0	18	0	16	
4	18	0	1	4	0	
5	1	0	4	0	8	
6	8	17	18	10	0	
7	0	4	0	24	0	
8	8	0	16	2	20	
9	2	13	0	16	0	
10	0	12	0	1	0	
11	0	9	20	24	0	
12	0	10	0	0	0	
13	0	28	0	0	16	
14	5	10	16	0	3	
15	24	0	0	18	8	

	Kemik Erimesi	YOK
0	24	0
1	0	9
2	12	0
3	0	4
4	4	0
5	16	2
6	0	4
7	0	0
8	0	1
9	1	0
10	8	0
11	1	0
12	0	1
13	8	3
14	0	0
15	2	1

[16 rows x 31 columns]

```
[36]: # DataFrame'i Excel dosyasına kaydet
df.to_excel("deneme.xlsx", index=False)
```

```
[37]: df = df.drop(['Kronik Hastaliklarim',
                    'Baba Kronik Hastaliklari',
                    'Anne Kronik Hastaliklari',
                    'Kiz Kardes Kronik Hastaliklari',
                    'Erkek Kardes Kronik Hastaliklari'], axis=1)
df.head()
```

```
[37]:
```

	Kullanici_id	Cinsiyet	Dogum_Tarihi	Uyruk	Il \
0	107	Male	1960-03-01	Turkiye	Canakkale
1	140	Male	1939-10-12	Turkiye	Trabzon
2	2	Female	1976-12-17	Turkiye	Canakkale
3	83	Male	1977-06-17	Turkiye	Adana
4	7	Female	1976-09-03	Turkiye	Izmir

	Ilac_Adi	Ilac_Baslangic_Tarihi	Ilac_Bitis_Tarihi \
0	trifluoperazine	2022-01-09	2022-03-04
1	fluphenazine hcl	2022-01-09	2022-03-08
2	warfarin sodium	2022-01-11	2022-03-12
3	valproic acid	2022-01-04	2022-03-12
4	carbamazepine extended release	2022-01-13	2022-03-06

	Yan_Etki	Yan_Etki_Bildirim_Tarihi	... Diger	Diyabet	Guatr \
0	Kabizlik	2022-02-19 18:28:43	...	0	0 18
1	Yorgunluk	2022-02-03 20:48:17	...	2	16 2

2	Carpinti	2022-02-04 05:29:20 ...	16	15	0
3	Sinirlilik	2022-02-08 01:01:21 ...	3	1	0
4	Agizda Farkli Bir Tat	2022-02-12 05:33:06 ...	8	9	0

	Hipertansiyon	KOAH	Kalp Hastaliklari	Kan Hastaliklari	Kanser	\
0	3	4	8	1	0	
1	4	16	4	0	0	
2	0	2	1	0	0	
3	0	0	18	0	16	
4	18	0	1	4	0	

	Kemik Erimesi	YOK
0	24	0
1	0	9
2	12	0
3	0	4
4	4	0

[5 rows x 26 columns]

[38]: df.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2357 entries, 0 to 2356
Data columns (total 26 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Kullanici_id                         2357 non-null   int64
1   Cinsiyet                             1579 non-null   category
2   Dogum_Tarihi                         2357 non-null   datetime64[ns]
3   Uyruk                                2357 non-null   category
4   Il                                    2130 non-null   category
5   Ilac_Adi                             2357 non-null   category
6   Ilac_Baslangic_Tarihi                2357 non-null   datetime64[ns]
7   Ilac_Bitis_Tarihi                    2357 non-null   datetime64[ns]
8   Yan_Etki                             2357 non-null   category
9   Yan_Etki_Bildirim_Tarihi             2357 non-null   datetime64[ns]
10  Alerjilerim                           1873 non-null   category
11  Kan_Grubu                             2010 non-null   category
12  Kilo                                  2064 non-null   float64
13  Boy                                   2243 non-null   float64
14  Alzheimer                             2357 non-null   int64
15  Astim                                 2357 non-null   int64
16  Diger                                 2357 non-null   int64
17  Diyabet                               2357 non-null   int64
18  Guatr                                 2357 non-null   int64
19  Hipertansiyon                         2357 non-null   int64
```

```

20 KOAH                2357 non-null    int64
21 Kalp Hastaliklari   2357 non-null    int64
22 Kan Hastaliklari    2357 non-null    int64
23 Kanser              2357 non-null    int64
24 Kemik Erimesi       2357 non-null    int64
25 YOK                 2357 non-null    int64
dtypes: category(7), datetime64[ns](4), float64(2), int64(13)
memory usage: 376.9 KB

```

```
[39]: df.isnull().sum()
```

```

[39]: Kullanici_id      0
      Cinsiyet          778
      Dogum_Tarihi      0
      Uyruk             0
      Il               227
      Ilac_Adi          0
      Ilac_Baslangic_Tarihi  0
      Ilac_Bitis_Tarihi  0
      Yan_Etki          0
      Yan_Etki_Bildirim_Tarihi  0
      Alerjilerim       484
      Kan_Grubu         347
      Kilo              293
      Boy              114
      Alzheimer         0
      Astim             0
      Diger             0
      Diyabet           0
      Guatr             0
      Hipertansiyon     0
      KOAH              0
      Kalp_Hastaliklari  0
      Kan_Hastaliklari  0
      Kanser            0
      Kemik_Erimesi     0
      YOK               0
      dtype: int64

```

```
[40]: df.describe().T #eksik gözlemleri göz ardı eder ve kategorik değişkenleri ↵
      ↪görmezden gelir
```

```

[40]:
      count      mean \
Kullanici_id      2357.0      97.216801
Dogum_Tarihi      2357    1974-11-25 04:06:12.677131936
Ilac_Baslangic_Tarihi  2357    2022-01-07 10:47:36.173101312
Ilac_Bitis_Tarihi   2357    2022-03-10 16:25:27.365294848

```

Yan_Etki_Bildirim_Tarihi	2357	2022-02-10 17:09:30.742044928
Kilo	2064.0	80.863857
Boy	2243.0	174.638431
Alzheimer	2357.0	5.290199
Astim	2357.0	4.811201
Diger	2357.0	2.571065
Diyabet	2357.0	4.742469
Guatr	2357.0	4.174374
Hipertansiyon	2357.0	3.818838
KOAH	2357.0	4.42045
Kalp Hastaliklari	2357.0	4.812473
Kan Hastaliklari	2357.0	5.0
Kanser	2357.0	4.700467
Kemik Erimesi	2357.0	5.750106
YOK	2357.0	3.467968

		min	25%	\
Kullanici_id		1.0	47.0	
Dogum_Tarihi	1939-10-12 00:00:00	1959-02-05 00:00:00		
Ilac_Baslangic_Tarihi	2022-01-01 00:00:00	2022-01-04 00:00:00		
Ilac_Bitis_Tarihi	2022-03-02 00:00:00	2022-03-06 00:00:00		
Yan_Etki_Bildirim_Tarihi	2022-02-01 04:34:33	2022-02-04 05:29:20		
Kilo	50.0	65.0		
Boy	145.0	160.0		
Alzheimer	0.0	0.0		
Astim	0.0	0.0		
Diger	0.0	0.0		
Diyabet	0.0	0.0		
Guatr	0.0	0.0		
Hipertansiyon	0.0	0.0		
KOAH	0.0	0.0		
Kalp Hastaliklari	0.0	0.0		
Kan Hastaliklari	0.0	0.0		
Kanser	0.0	0.0		
Kemik Erimesi	0.0	0.0		
YOK	0.0	0.0		

		50%	75%	\
Kullanici_id		97.0	146.0	
Dogum_Tarihi	1973-09-09 00:00:00	1992-03-24 00:00:00		
Ilac_Baslangic_Tarihi	2022-01-07 00:00:00	2022-01-11 00:00:00		
Ilac_Bitis_Tarihi	2022-03-11 00:00:00	2022-03-15 00:00:00		
Yan_Etki_Bildirim_Tarihi	2022-02-09 20:53:54	2022-02-17 07:08:01		
Kilo	83.0	96.0		
Boy	176.0	187.0		
Alzheimer	2.0	9.0		
Astim	1.0	8.0		

Diger	0.0	2.0
Diyabet	1.0	8.0
Guatr	0.0	6.0
Hipertansiyon	1.0	5.0
KOAH	1.0	6.0
Kalp Hastaliklari	1.0	8.0
Kan Hastaliklari	1.0	8.0
Kanser	2.0	8.0
Kemik Erimesi	2.0	10.0
YOK	0.0	4.0

	max	std
Kullanici_id	196.0	57.0172
Dogum_Tarihi	2011-04-25 00:00:00	NaN
Ilac_Baslangic_Tarihi	2022-01-14 00:00:00	NaN
Ilac_Bitis_Tarihi	2022-03-19 00:00:00	NaN
Yan_Etki_Bildirim_Tarihi	2022-02-19 21:47:39	NaN
Kilo	110.0	18.635269
Boy	203.0	16.516552
Alzheimer	25.0	6.52745
Astim	31.0	6.759668
Diger	24.0	5.194328
Diyabet	26.0	6.610709
Guatr	26.0	6.815957
Hipertansiyon	28.0	6.005361
KOAH	28.0	6.751732
Kalp Hastaliklari	30.0	6.885281
Kan Hastaliklari	28.0	7.469747
Kanser	26.0	6.580845
Kemik Erimesi	26.0	7.430602
YOK	25.0	5.843344

```
[41]: def sutun_istatistikleri(df, sutun_adi):
    print(f"Ortalama: {df[sutun_adi].mean()}")
    print(f"Dolu Gözlem Sayısı: {df[sutun_adi].count()}")
    print(f"Maksimum Değer: {df[sutun_adi].max()}")
    print(f"Minimum Değer: {df[sutun_adi].min()}")
    print(f"Medyan: {df[sutun_adi].median()}")
    print(f"Standart Sapma: {df[sutun_adi].std()}")
```

```
[42]: sutun_istatistikleri(df, "Kilo")
```

```
Ortalama: 80.86385658914729
Dolu Gözlem Sayısı: 2064
Maksimum Değer: 110.0
Minimum Değer: 50.0
Medyan: 83.0
```

Standart Sapma: 18.635269357691442

```
[43]: sutun_istatistikleri(df, "Boy")
```

Ortalama: 174.63843067320553
Dolu Gözlem Sayısı: 2243
Maksimum Değer: 203.0
Minimum Değer: 145.0
Medyan: 176.0
Standart Sapma: 16.516552496435665

Görselleştirme df.info()

```
[44]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2357 entries, 0 to 2356
Data columns (total 26 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Kullanici_id          2357 non-null   int64
1   Cinsiyet              1579 non-null   category
2   Dogum_Tarihi          2357 non-null   datetime64[ns]
3   Uyruk                 2357 non-null   category
4   Il                    2130 non-null   category
5   Ilac_Adi              2357 non-null   category
6   Ilac_Baslangic_Tarihi 2357 non-null   datetime64[ns]
7   Ilac_Bitis_Tarihi     2357 non-null   datetime64[ns]
8   Yan_Etki              2357 non-null   category
9   Yan_Etki_Bildirim_Tarihi 2357 non-null   datetime64[ns]
10  Alerjilerim           1873 non-null   category
11  Kan_Grubu             2010 non-null   category
12  Kilo                  2064 non-null   float64
13  Boy                   2243 non-null   float64
14  Alzheimer             2357 non-null   int64
15  Astim                 2357 non-null   int64
16  Diger                 2357 non-null   int64
17  Diyabet               2357 non-null   int64
18  Guatr                 2357 non-null   int64
19  Hipertansiyon         2357 non-null   int64
20  KOAH                  2357 non-null   int64
21  Kalp_Hastaliklari     2357 non-null   int64
22  Kan_Hastaliklari      2357 non-null   int64
23  Kanser                2357 non-null   int64
24  Kemik_Erimesi         2357 non-null   int64
25  YOK                   2357 non-null   int64
dtypes: category(7), datetime64[ns](4), float64(2), int64(13)
memory usage: 376.9 KB
```


0.0.3 Cinsiyete göre ilaç kullanım sayısı

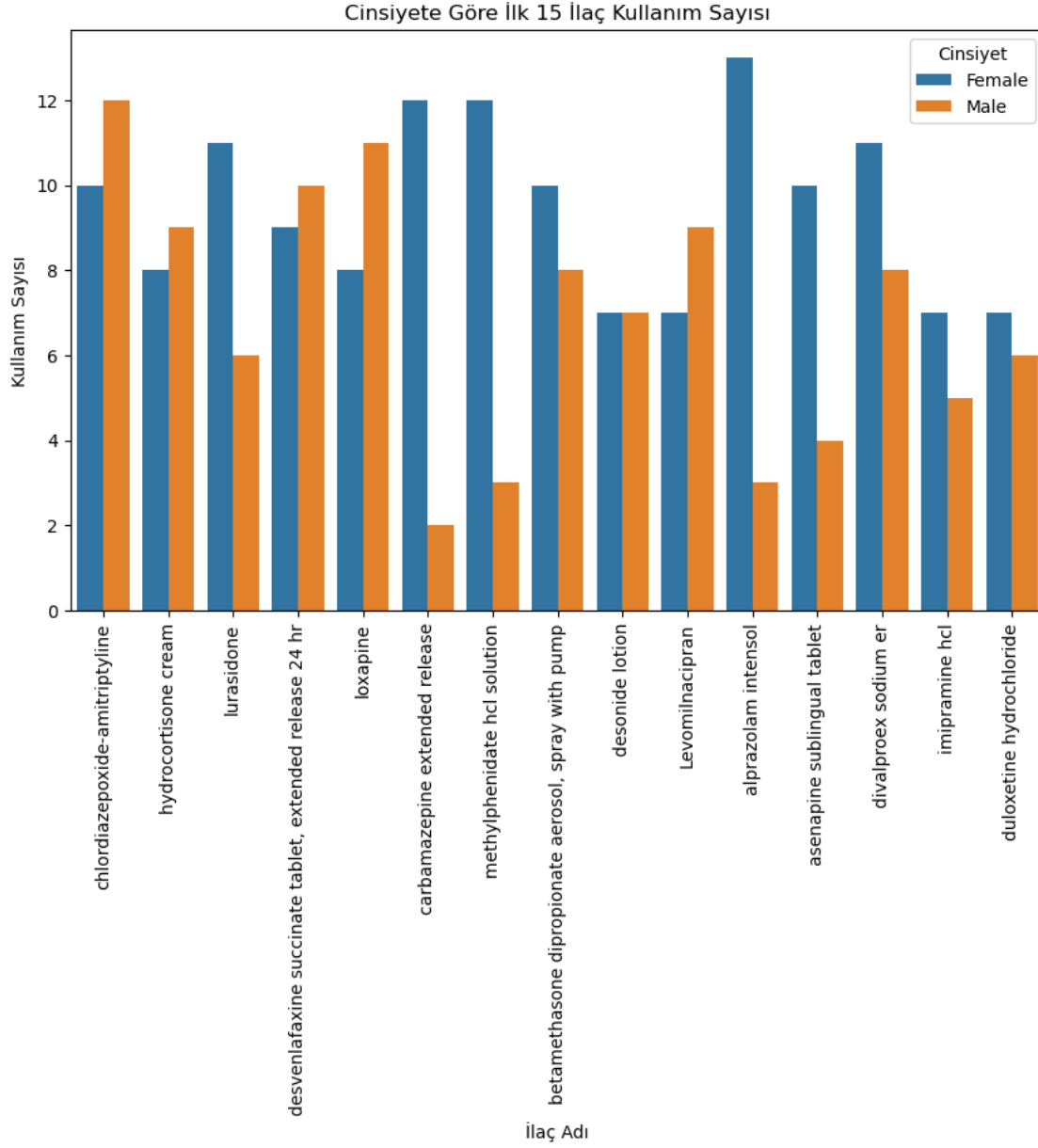
```
[46]: import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# En çok kullanılan 15 ilacı belirle
top_15_ilac = df['Ilac_Adi'].value_counts().nlargest(15).index

# Filtreleme
df_top_15 = df[df['Ilac_Adi'].isin(top_15_ilac)]

# Barplot oluştur
plt.figure(figsize=(10, 6))
sns.countplot(data=df_top_15, x='Ilac_Adi', hue='Cinsiyet', order=top_15_ilac)
plt.xticks(rotation=90)
plt.title('Cinsiyete Göre İlk 15 İlaç Kullanım Sayısı')
plt.xlabel('İlaç Adı')
plt.ylabel('Kullanım Sayısı')
plt.legend(title='Cinsiyet')
plt.show()
```

```
C:\Users\ENIAC\anaconda3\Lib\site-packages\seaborn\categorical.py:641:
FutureWarning: The default of observed=False is deprecated and will be changed
to True in a future version of pandas. Pass observed=False to retain current
behavior or observed=True to adopt the future default and silence this warning.
    grouped_vals = vals.groupby(grouper)
```



0.0.4 Cinsiyet ve yan etki arasındaki ilişki

```
[47]: import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Catplot oluştur
plt.figure(figsize=(10, 6))
```

```
sns.catplot(data=df, x='Yan_Etki', hue='Cinsiyet', kind='count', height=6,
            aspect=2)
plt.xticks(rotation=90)
plt.title('Cinsiyete Göre Yan Etki Dağılımı')
plt.xlabel('Yan Etki')
plt.ylabel('Kullanım Sayısı')
plt.legend(title='Cinsiyet')
plt.show()
```

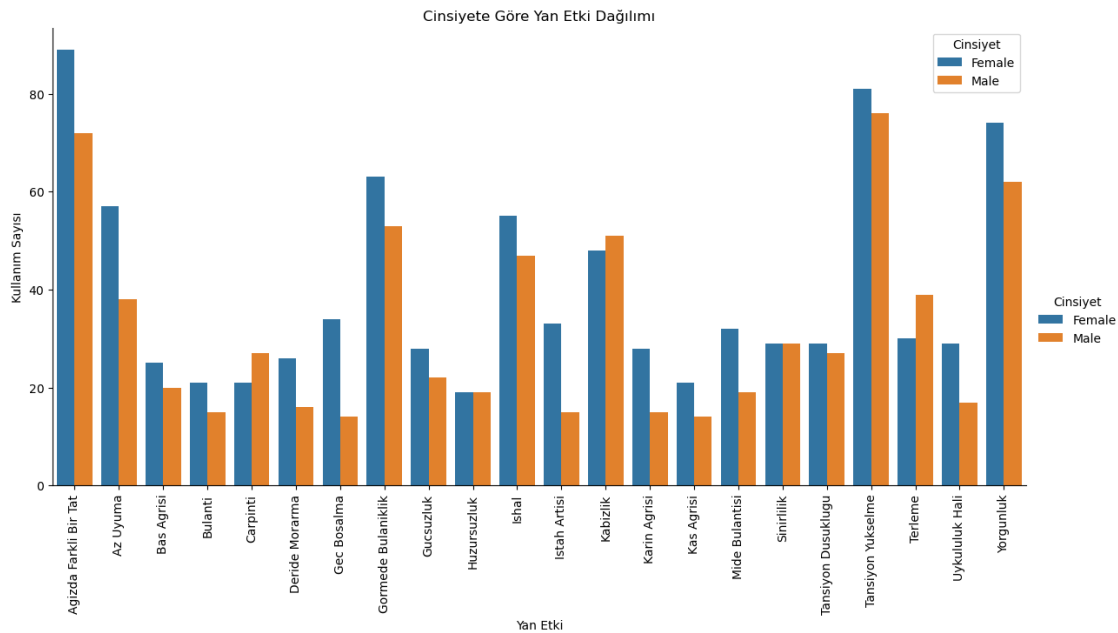
C:\Users\ENIAC\anaconda3\Lib\site-packages\seaborn\categorical.py:641:
FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.

grouped_vals = vals.groupby(grouper)

C:\Users\ENIAC\anaconda3\Lib\site-packages\seaborn\categorical.py:641:
FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.

grouped_vals = vals.groupby(grouper)

<Figure size 1000x600 with 0 Axes>



0.0.5 Cinsiyete göre ortalama kilo ve boy değerleri

```
[49]: import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Cinsiyete göre ortalama kilo ve boy değerlerini hesapla
grouped_data = df.groupby('Cinsiyet')[['Kilo', 'Boy']].mean().reset_index()

# Barplot oluştur
plt.figure(figsize=(12, 6))
sns.barplot(data=grouped_data.melt(id_vars='Cinsiyet', value_vars=['Kilo', 'Boy']),
            x='Cinsiyet', y='value', hue='variable')
plt.title('Cinsiyete Göre Ortalama Kilo ve Boy')
plt.xlabel('Cinsiyet')
plt.ylabel('Değer')
plt.legend(title='Ölçüm', labels=['Kilo', 'Boy'])
plt.show()
```

C:\Users\ENIAC\AppData\Local\Temp\ipykernel_9104\825038169.py:6: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.

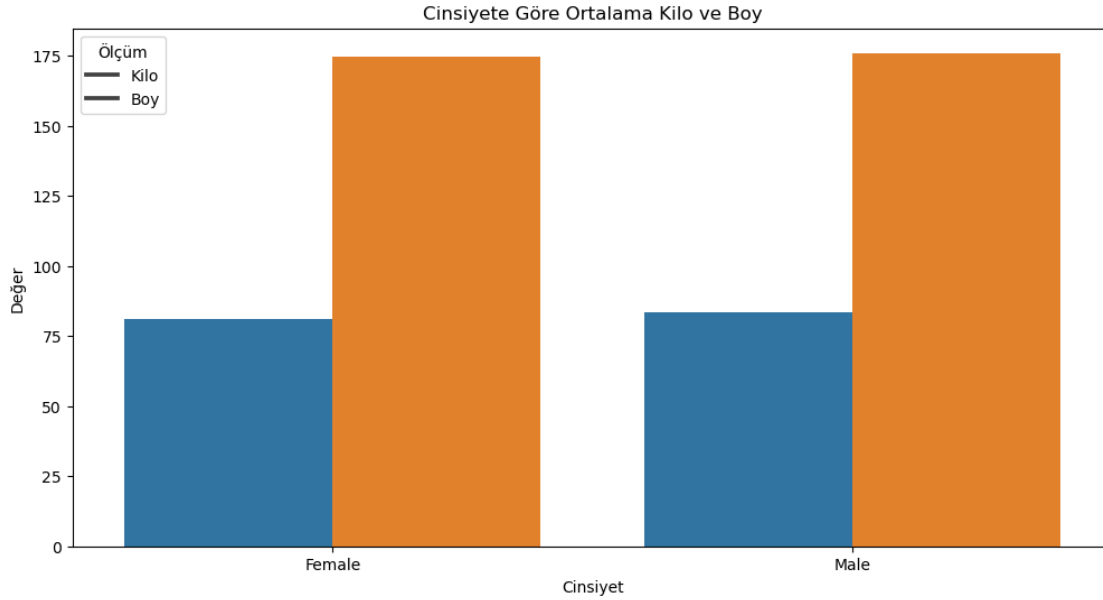
```
grouped_data = df.groupby('Cinsiyet')[['Kilo', 'Boy']].mean().reset_index()
```

C:\Users\ENIAC\anaconda3\Lib\site-packages\seaborn\categorical.py:641: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.

```
grouped_vals = vals.groupby(grouper)
```

C:\Users\ENIAC\anaconda3\Lib\site-packages\seaborn\categorical.py:641: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.

```
grouped_vals = vals.groupby(grouper)
```

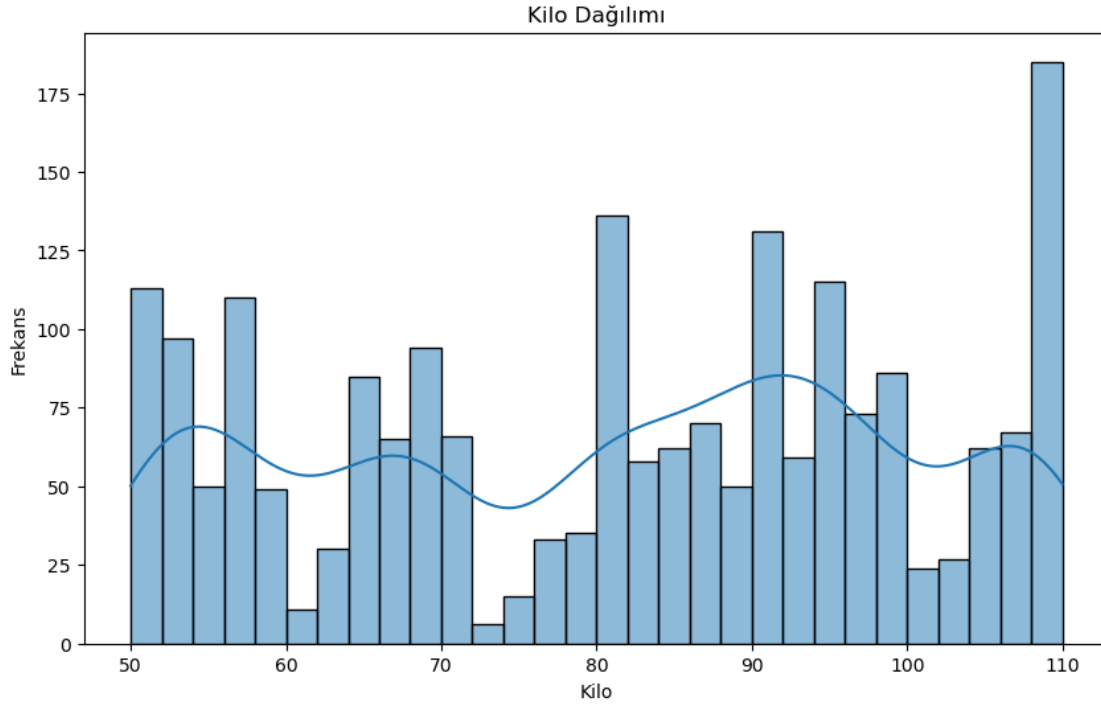


0.0.6 Kilo dağılımı

```
[50]: import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Kilo dağılımını gösteren histplot
plt.figure(figsize=(10, 6))
sns.histplot(data=df, x='Kilo', bins=30, kde=True)
plt.title('Kilo Dağılımı')
plt.xlabel('Kilo')
plt.ylabel('Frekans')
plt.show()
```

```
C:\Users\ENIAC\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed in a
future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
```

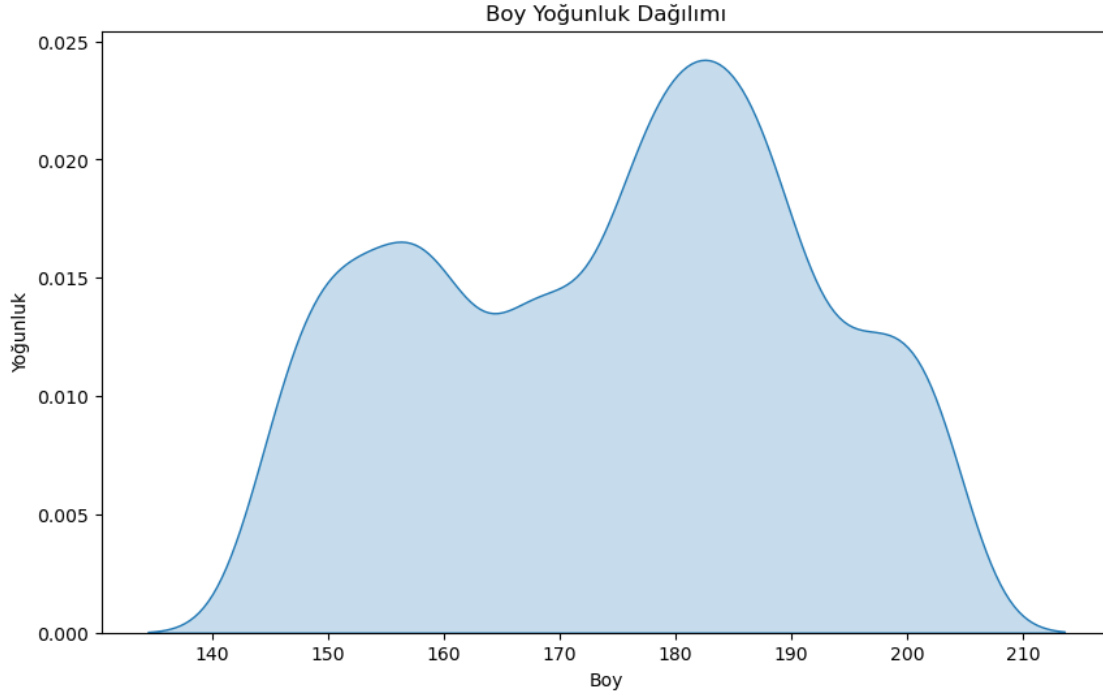


0.0.7 Boyun yoğunluk dağılımı

```
[51]: import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Boy yoğunluk dağılımını gösteren kdeplot
plt.figure(figsize=(10, 6))
sns.kdeplot(data=df['Boy'], fill=True)
plt.title('Boy Yoğunluk Dağılımı')
plt.xlabel('Boy')
plt.ylabel('Yoğunluk')
plt.show()
```

C:\Users\ENIAC\anaconda3\Lib\site-packages\seaborn_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed in a
future version. Convert inf values to NaN before operating instead.
with pd.option_context('mode.use_inf_as_na', True):

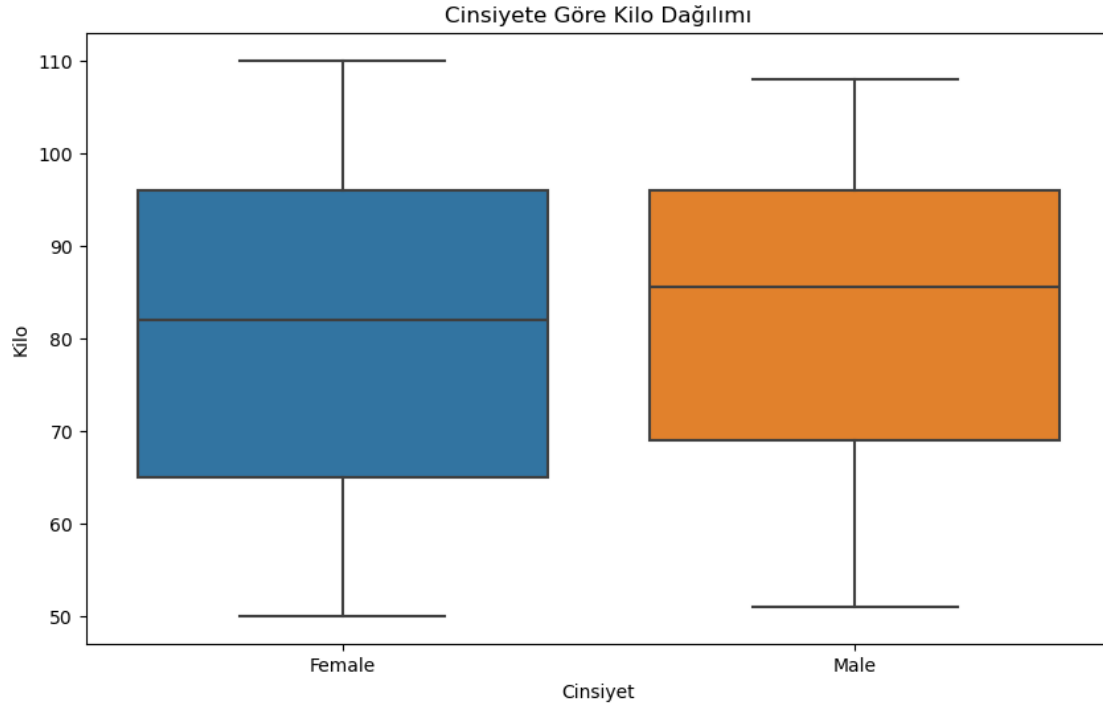


0.0.8 Cinsiyete göre kilo değerlerinin dağılımı

```
[52]: import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Cinsiyete göre kilo dağılımını gösteren boxplot
plt.figure(figsize=(10, 6))
sns.boxplot(data=df, x='Cinsiyet', y='Kilo')
plt.title('Cinsiyete Göre Kilo Dağılımı')
plt.xlabel('Cinsiyet')
plt.ylabel('Kilo')
plt.show()
```

```
C:\Users\ENIAC\anaconda3\Lib\site-packages\seaborn\categorical.py:641:
FutureWarning: The default of observed=False is deprecated and will be changed
to True in a future version of pandas. Pass observed=False to retain current
behavior or observed=True to adopt the future default and silence this warning.
  grouped_vals = vals.groupby(grouper)
```

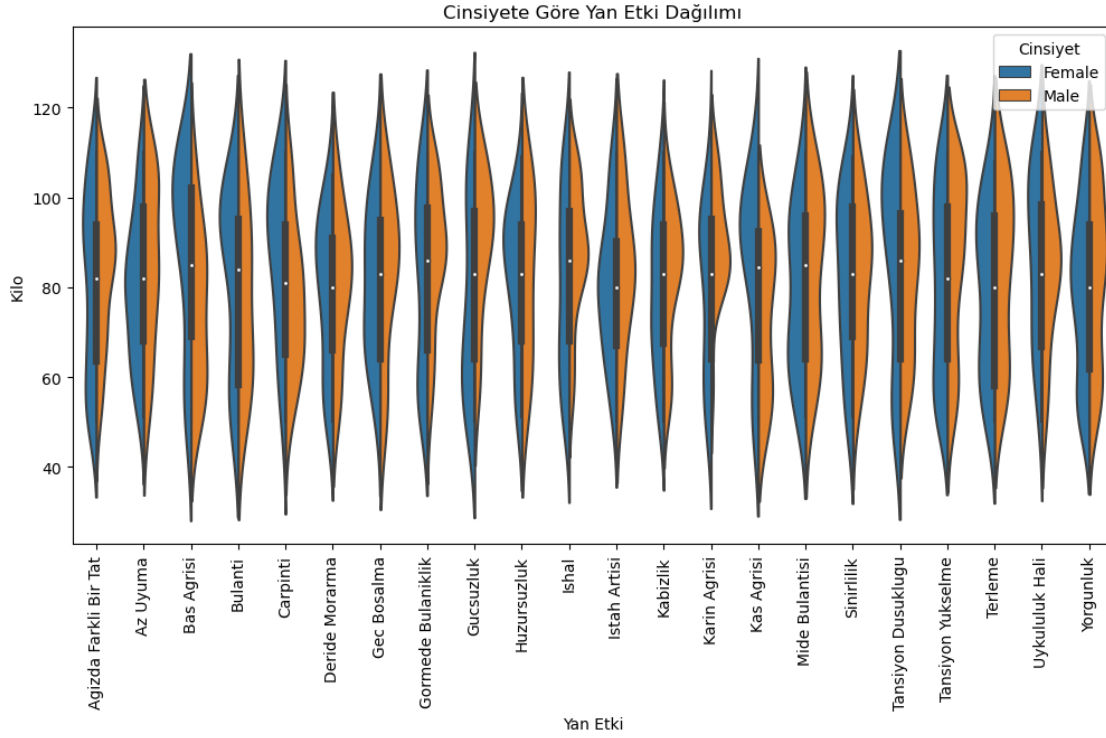


0.0.9 Cinsiyete göre yan etkilerin dağılımı

```
[53]: import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Cinsiyete göre yan etki dağılımını gösteren violinplot
plt.figure(figsize=(12, 6))
sns.violinplot(data=df, x='Yan_Etki', y='Kilo', hue='Cinsiyet', split=True)
plt.xticks(rotation=90)
plt.title('Cinsiyete Göre Yan Etki Dağılımı')
plt.xlabel('Yan Etki')
plt.ylabel('Kilo')
plt.legend(title='Cinsiyet')
plt.show()
```

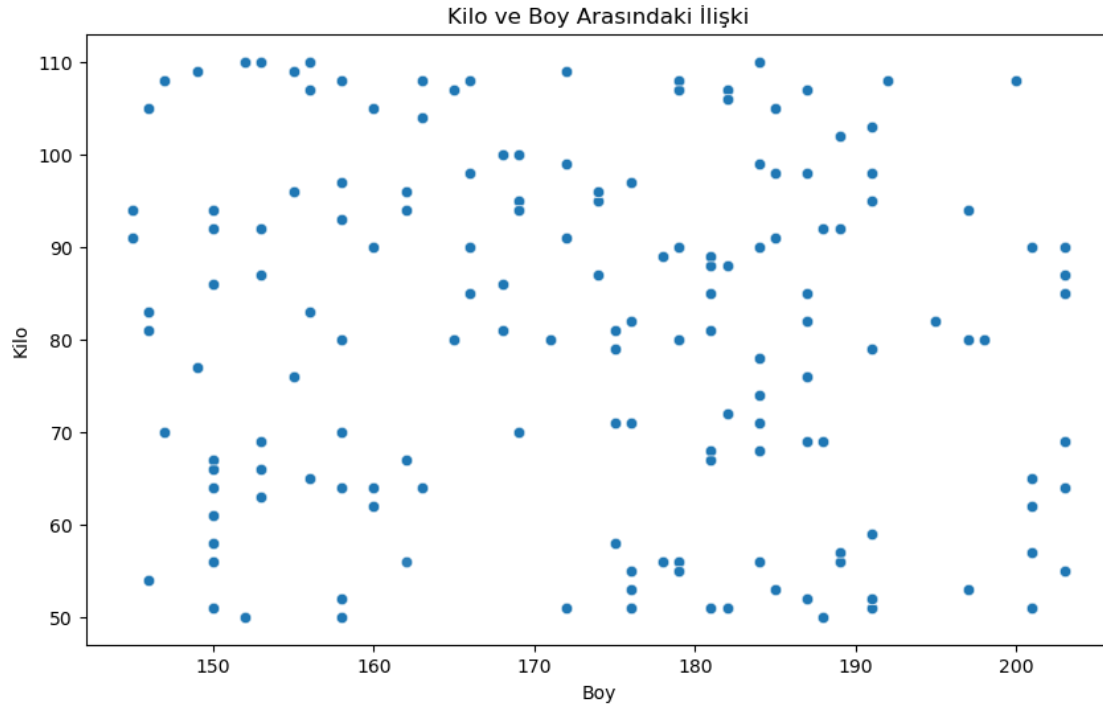
```
C:\Users\ENIAC\anaconda3\Lib\site-packages\seaborn\categorical.py:641:
FutureWarning: The default of observed=False is deprecated and will be changed
to True in a future version of pandas. Pass observed=False to retain current
behavior or observed=True to adopt the future default and silence this warning.
    grouped_vals = vals.groupby(grouper)
```

0.0.10 Kilo ve boy arasındaki ilişki

```
[54]: import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Kilo ve boy arasındaki ilişkiyi gösteren scatter plot
plt.figure(figsize=(10, 6))
sns.scatterplot(data=df, x='Boy', y='Kilo', alpha=0.6)
plt.title('Kilo ve Boy Arasındaki İlişki')
plt.xlabel('Boy')
plt.ylabel('Kilo')
plt.show()
```

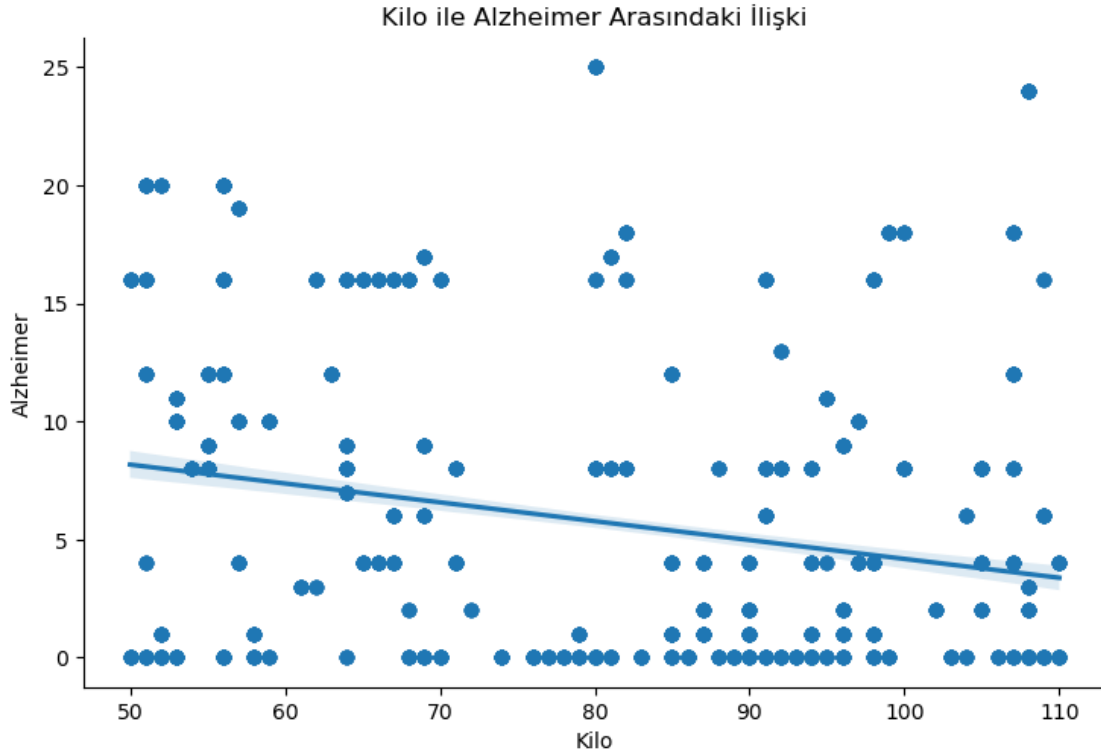


0.0.11 Kilo ile Alzheimer arasındaki ilişkiyi

```
[55]: import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Kilo ile Alzheimer arasındaki ilişkiyi gösteren lmlot
plt.figure(figsize=(10, 6))
sns.lmplot(data=df, x='Kilo', y='Alzheimer', aspect=1.5)
plt.title('Kilo ile Alzheimer Arasındaki İlişki')
plt.xlabel('Kilo')
plt.ylabel('Alzheimer')
plt.show()
```

<Figure size 1000x600 with 0 Axes>



0.0.12 Tüm sayısal değişkenler arasındaki ilişki

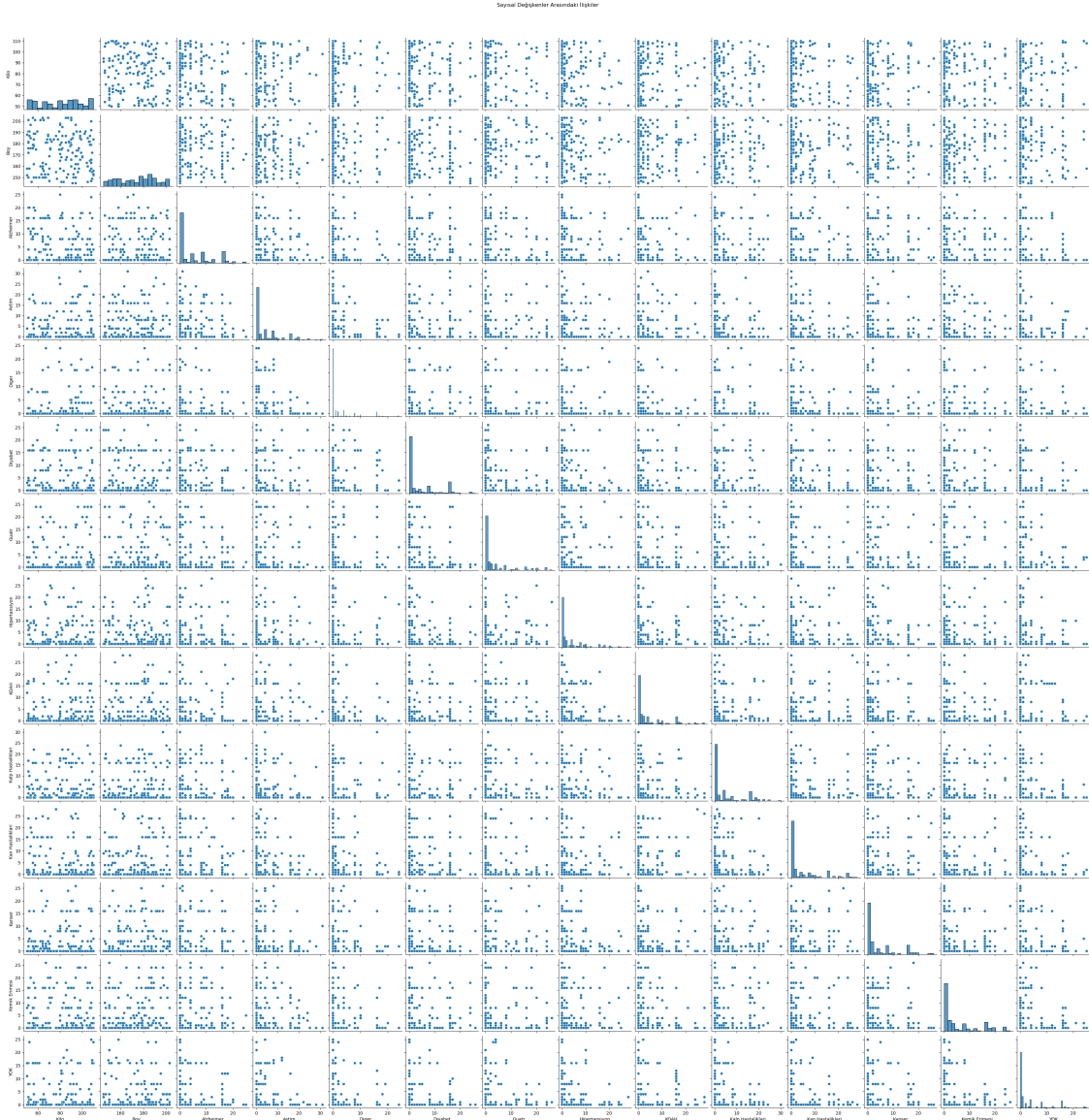
```
[56]: import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Sayısal değişkenler için pairplot oluştur
num_cols = ['Kilo', 'Boy', 'Alzheimer', 'Astim', 'Diger', 'Diyabet', 'Guatr',
            'Hipertansiyon', 'KOA', 'Kalp Hastalıkları', 'Kan Hastalıkları',
            'Kanser', 'Kemik Erimesi', 'YOK']

sns.pairplot(df[num_cols])
plt.suptitle('Sayısal Değişkenler Arasındaki İlişkiler', y=1.02)
plt.show()
```

C:\Users\ENIAC\anaconda3\Lib\site-packages\seaborn_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed in a
future version. Convert inf values to NaN before operating instead.
with pd.option_context('mode.use_inf_as_na', True):
C:\Users\ENIAC\anaconda3\Lib\site-packages\seaborn_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed in a
future version. Convert inf values to NaN before operating instead.


```
with pd.option_context('mode.use_inf_as_na', True):
```



0.0.13 Korelasyon matrisi

```
[62]: import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Sayısal değişkenleri seç
num_cols = ['Kilo', 'Boy']

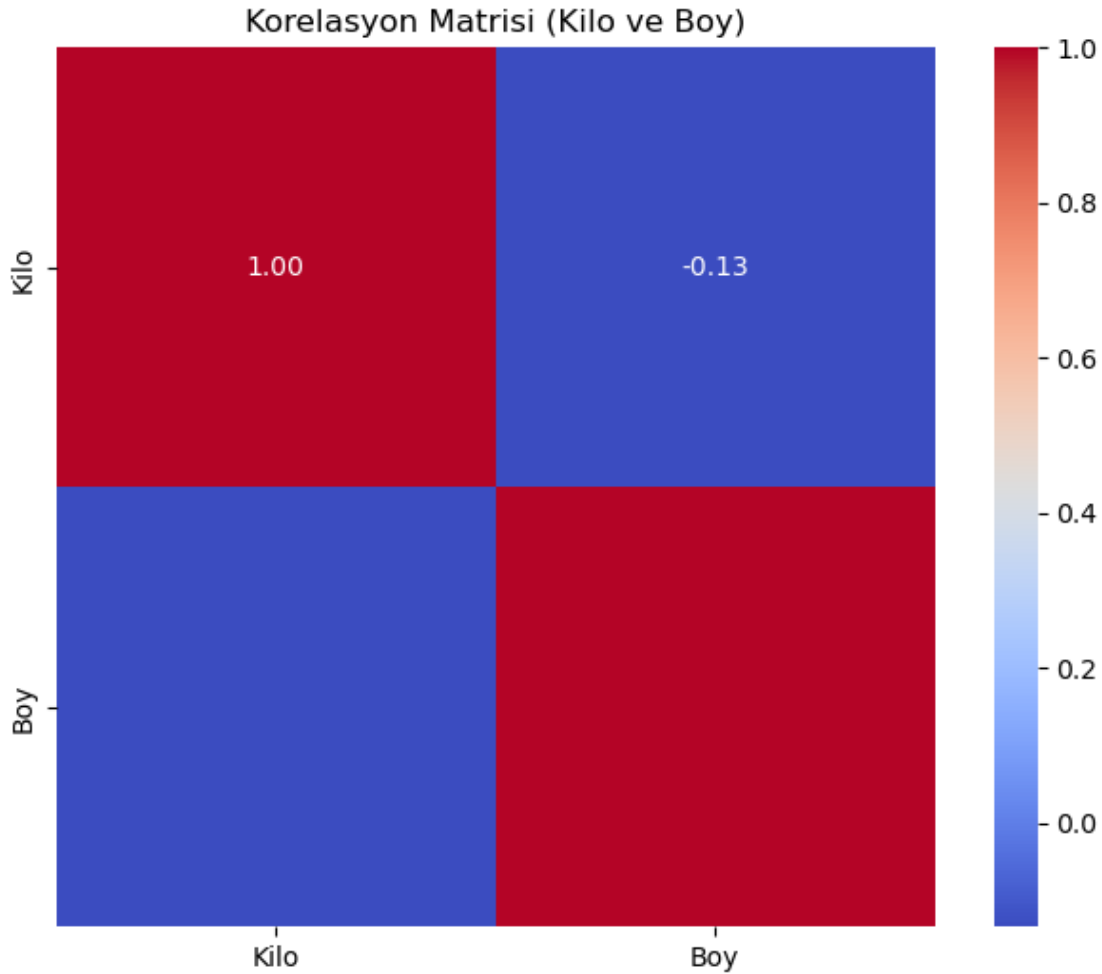
# Korelasyon matrisini hesapla
```

```

correlation_matrix = df[num_cols].corr()

# Heatmap oluştur
plt.figure(figsize=(8, 6))
sns.heatmap(correlation_matrix, annot=True, fmt=".2f", cmap='coolwarm',
            square=True)
plt.title('Korelasyon Matrisi (Kilo ve Boy)')
plt.show()

```



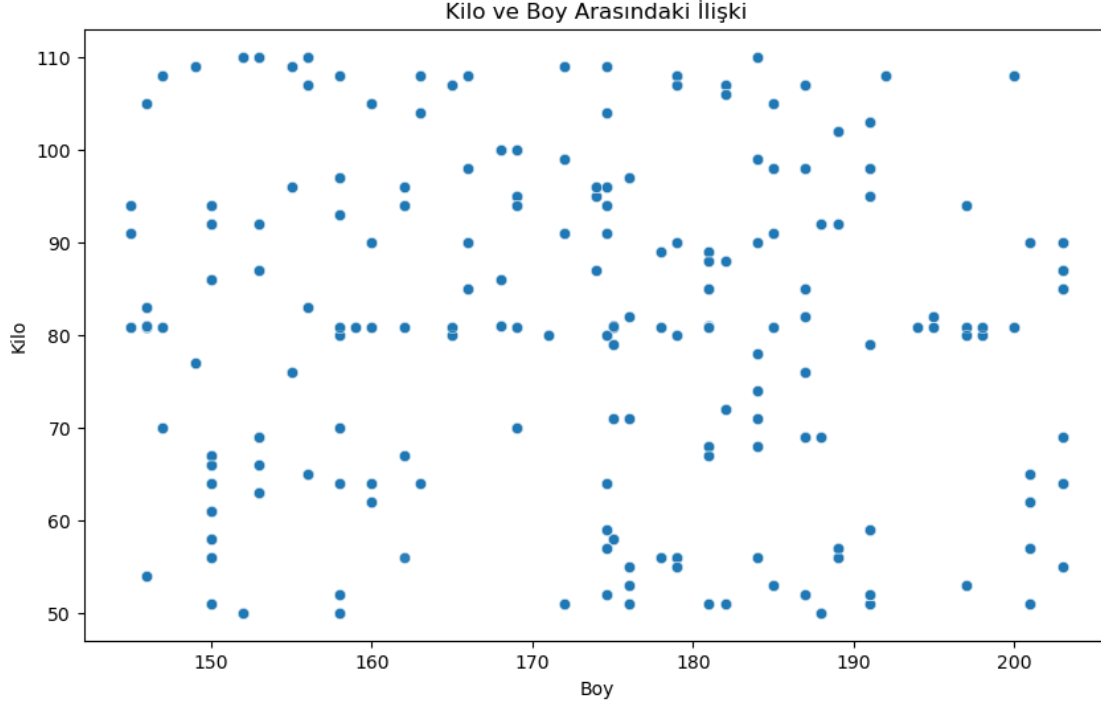
```

[63]: import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Kilo ve boy arasındaki ilişkiyi gösteren scatter plot
plt.figure(figsize=(10, 6))
sns.scatterplot(data=df, x='Boy', y='Kilo', alpha=0.6)

```

```
plt.title('Kilo ve Boy Arasındaki İlişki')
plt.xlabel('Boy')
plt.ylabel('Kilo')
plt.show()
```



Veri Ön İşleme eksik değer doldurma aşaması

```
[64]: df.head()
```

```
[64]:
```

	Kullanici_id	Cinsiyet	Dogum_Tarihi	Uyruk	Il	\
0	107	1.0	1960-03-01	Turkiye	Canakkale	
1	140	1.0	1939-10-12	Turkiye	Trabzon	
2	2	0.0	1976-12-17	Turkiye	Canakkale	
3	83	1.0	1977-06-17	Turkiye	Adana	
4	7	0.0	1976-09-03	Turkiye	Izmir	

	Ilac_Adi	Ilac_Baslangic_Tarihi	Ilac_Bitis_Tarihi	\
0	trifluoperazine	2022-01-09	2022-03-04	
1	fluphenazine hcl	2022-01-09	2022-03-08	
2	warfarin sodium	2022-01-11	2022-03-12	
3	valproic acid	2022-01-04	2022-03-12	
4	carbamazepine extended release	2022-01-13	2022-03-06	

	Yan_Etki	Yan_Etki_Bildirim_Tarihi	... Diger	Diyabet	Guatr	\

0	Kabizlik	2022-02-19 18:28:43	...	0.0	0.0	18.0
1	Yorgunluk	2022-02-03 20:48:17	...	2.0	16.0	2.0
2	Carpinti	2022-02-04 05:29:20	...	16.0	15.0	0.0
3	Sinirlilik	2022-02-08 01:01:21	...	3.0	1.0	0.0
4	Agizda Farkli Bir Tat	2022-02-12 05:33:06	...	8.0	9.0	0.0

	Hipertansiyon	KOAH	Kalp Hastaliklari	Kan Hastaliklari	Kanser	\
0	3.0	4.0	8.0	1.0	0.0	
1	4.0	16.0	4.0	0.0	0.0	
2	0.0	2.0	1.0	0.0	0.0	
3	0.0	0.0	18.0	0.0	16.0	
4	18.0	0.0	1.0	4.0	0.0	

	Kemik Erimesi	YOK
0	24.0	0.0
1	0.0	9.0
2	12.0	0.0
3	0.0	4.0
4	4.0	0.0

[5 rows x 26 columns]

[65]: df.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2357 entries, 0 to 2356
Data columns (total 26 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Kullanici_id                          2357 non-null   int64
1   Cinsiyet                              2357 non-null   float64
2   Dogum_Tarihi                          2357 non-null   datetime64[ns]
3   Uyruk                                  2357 non-null   category
4   Il                                      2130 non-null   category
5   Ilac_Adi                              2357 non-null   category
6   Ilac_Baslangic_Tarihi                 2357 non-null   datetime64[ns]
7   Ilac_Bitis_Tarihi                     2357 non-null   datetime64[ns]
8   Yan_Etki                              2357 non-null   category
9   Yan_Etki_Bildirim_Tarihi              2357 non-null   datetime64[ns]
10  Alerjilerim                            1873 non-null   category
11  Kan_Grubu                              2357 non-null   float64
12  Kilo                                    2357 non-null   float64
13  Boy                                     2357 non-null   float64
14  Alzheimer                              2357 non-null   float64
15  Astim                                  2357 non-null   float64
16  Diger                                  2357 non-null   float64
17  Diyabet                                2357 non-null   float64
```



```

18  Guatr                2357 non-null  float64
19  Hipertansiyon        2357 non-null  float64
20  KOAH                 2357 non-null  float64
21  Kalp Hastaliklari    2357 non-null  float64
22  Kan Hastaliklari     2357 non-null  float64
23  Kanser               2357 non-null  float64
24  Kemik Erimesi        2357 non-null  float64
25  YOK                  2357 non-null  float64
dtypes: category(5), datetime64[ns](4), float64(16), int64(1)
memory usage: 408.6 KB

```

```
[66]: df.isnull().sum()
```

```

[66]: Kullanici_id      0
      Cinsiyet           0
      Dogum_Tarihi      0
      Uyruk             0
      Il                227
      Ilac_Adi           0
      Ilac_Baslangic_Tarihi 0
      Ilac_Bitis_Tarihi  0
      Yan_Etki           0
      Yan_Etki_Bildirim_Tarihi 0
      Alerjilerim       484
      Kan_Grubu          0
      Kilo              0
      Boy               0
      Alzheimer          0
      Astim              0
      Diger              0
      Diyabet            0
      Guatr              0
      Hipertansiyon      0
      KOAH               0
      Kalp_Hastaliklari  0
      Kan_Hastaliklari   0
      Kanser             0
      Kemik_Erimesi      0
      YOK                0
dtype: int64

```

0.0.14 Eksik Değerlerin İşlenmesi

Cinsiyet: En sık görülen değerle doldur. Il: En sık görülen değerle doldur. Alerjilerim: En sık görülen değerle doldur. Kan Grubu: En sık görülen değerle doldur. Kilo: Ortalama ile doldur. Boy: Ortalama ile doldur.

OneHotEncoder ile Cinsiyet, Il, Alerjilerim ve Kan Grubu'nu kodla.

Kilo ve Boy için StandardScaler kullan.

```
[67]: from sklearn.impute import SimpleImputer

      # Cinsiyet sütununu en sık görülen değerle doldurma
      imputer = SimpleImputer(strategy='most_frequent')
      df['Cinsiyet'] = imputer.fit_transform(df[['Cinsiyet']]).ravel()

[68]: # Il sütununu en sık görülen değerle doldurma
      imputer = SimpleImputer(strategy='most_frequent')
      df['Il'] = imputer.fit_transform(df[['Il']]).ravel()

[69]: # Alerjilerim sütununu en sık görülen değerle doldurma
      imputer = SimpleImputer(strategy='most_frequent')
      df['Alerjilerim'] = imputer.fit_transform(df[['Alerjilerim']]).ravel()

[70]: # Kan Grubu sütununu en sık görülen değerle doldurma
      imputer = SimpleImputer(strategy='most_frequent')
      df['Kan Grubu'] = imputer.fit_transform(df[['Kan Grubu']]).ravel()

[71]: # Kilo sütununu ortalama ile doldurma
      imputer = SimpleImputer(strategy='mean')
      df['Kilo'] = imputer.fit_transform(df[['Kilo']]).ravel()

[72]: # Boy sütununu ortalama ile doldurma
      imputer = SimpleImputer(strategy='mean')
      df['Boy'] = imputer.fit_transform(df[['Boy']]).ravel()

[73]: df.isnull().sum()
```

```
[73]: Kullanici_id      0
      Cinsiyet          0
      Dogum_Tarihi      0
      Uyruk             0
      Il                0
      Ilac_Adi          0
      Ilac_Baslangic_Tarihi 0
      Ilac_Bitis_Tarihi  0
      Yan_Etki          0
      Yan_Etki_Bildirim_Tarihi 0
      Alerjilerim       0
      Kan_Grubu         0
      Kilo              0
      Boy              0
      Alzheimer         0
      Astim             0
      Diger             0
      Diyabet           0
```

```

Guatr                                0
Hipertansiyon                       0
KOAHA                               0
Kalp Hastaliklari                   0
Kan Hastaliklari                     0
Kanser                              0
Kemik Erimesi                       0
YOK                                  0
dtype: int64

```

```

[74]: from sklearn.preprocessing import LabelEncoder
label_encoder = LabelEncoder()
df['Cinsiyet'] = label_encoder.fit_transform(df['Cinsiyet'])
df['Kan Grubu'] = label_encoder.fit_transform(df['Kan Grubu'])

```

```

[75]: df.head()

```

```

[75]:  Kullanici_id  Cinsiyet  Dogum_Tarihi    Uyruk      Il  \
0           107         2   1960-03-01  Turkiye  Canakkale
1           140         2   1939-10-12  Turkiye   Trabzon
2             2         1   1976-12-17  Turkiye  Canakkale
3            83         2   1977-06-17  Turkiye    Adana
4             7         1   1976-09-03  Turkiye    Izmir

      Ilac_Adi  Ilac_Baslangic_Tarihi  Ilac_Bitis_Tarihi  \
0      trifluoperazine             2022-01-09          2022-03-04
1      fluphenazine hcl             2022-01-09          2022-03-08
2      warfarin sodium             2022-01-11          2022-03-12
3      valproic acid              2022-01-04          2022-03-12
4  carbamazepine extended release      2022-01-13          2022-03-06

      Yan_Etki  Yan_Etki_Bildirim_Tarihi  ...  Diger  Diyabet  Guatr  \
0      Kabizlik          2022-02-19 18:28:43  ...   0.0      0.0   18.0
1    Yorgunluk          2022-02-03 20:48:17  ...   2.0     16.0    2.0
2    Carpinti          2022-02-04 05:29:20  ...  16.0     15.0    0.0
3  Sinirlilik          2022-02-08 01:01:21  ...   3.0      1.0    0.0
4  Agizda Farkli Bir Tat          2022-02-12 05:33:06  ...   8.0      9.0    0.0

      Hipertansiyon  KOAH  Kalp Hastaliklari  Kan Hastaliklari  Kanser  \
0             3.0    4.0              8.0              1.0      0.0
1             4.0   16.0              4.0              0.0      0.0
2             0.0    2.0              1.0              0.0      0.0
3             0.0    0.0             18.0              0.0     16.0
4            18.0    0.0              1.0              4.0      0.0

      Kemik Erimesi  YOK
0             24.0    0.0

```

```

1          0.0  9.0
2         12.0  0.0
3          0.0  4.0
4          4.0  0.0

```

[5 rows x 26 columns]

```

[76]: from sklearn.preprocessing import OneHotEncoder

# Kategorik değişkenler
categorical_columns = ['Uyruk', 'Il', 'Ilac_Adi', 'Yan_Etki', 'Alerjilerim']

# Yeni DataFrame oluşturma
encoded_dfs = []

for col in categorical_columns:
    encoder = OneHotEncoder(sparse_output=False, drop='first') # drop='first'
    ↪ile ilk kategoriyi düşürerek dummy değişken tuzağından kaçınıyoruz
    encoded = encoder.fit_transform(df[[col]])
    encoded_df = pd.DataFrame(encoded, columns=encoder.
    ↪get_feature_names_out([col]))
    encoded_dfs.append(encoded_df)

# Tüm encoded DataFrame'leri birleştirme
final_encoded_df = pd.concat(encoded_dfs, axis=1)

# Orijinal DataFrame'den kategorik sütunları çıkarma ve yeni encoded DataFrame
↪ile birleştirme
df = pd.concat([df.drop(categorical_columns, axis=1), final_encoded_df], axis=1)

```

```

[77]: # Tarih değişkenlerinden yıl, ay ve gün çıkarma
df['Dogum_Yili'] = df['Dogum_Tarihi'].dt.year
df['Ilac_Baslangic_Yili'] = df['Ilac_Baslangic_Tarihi'].dt.year
df['Ilac_Bitis_Yili'] = df['Ilac_Bitis_Tarihi'].dt.year
df['Yan_Etki_Bildirim_Yili'] = df['Yan_Etki_Bildirim_Tarihi'].dt.year

df['Dogum_Ayi'] = df['Dogum_Tarihi'].dt.month
df['Ilac_Baslangic_Ayi'] = df['Ilac_Baslangic_Tarihi'].dt.month
df['Ilac_Bitis_Ayi'] = df['Ilac_Bitis_Tarihi'].dt.month
df['Yan_Etki_Bildirim_Ayi'] = df['Yan_Etki_Bildirim_Tarihi'].dt.month

df['Dogum_Gunu'] = df['Dogum_Tarihi'].dt.day
df['Ilac_Baslangic_Gunu'] = df['Ilac_Baslangic_Tarihi'].dt.day
df['Ilac_Bitis_Gunu'] = df['Ilac_Bitis_Tarihi'].dt.day
df['Yan_Etki_Bildirim_Gunu'] = df['Yan_Etki_Bildirim_Tarihi'].dt.day

```

```
[78]: # İlaç kullanım süresi (gün cinsinden)
df['Ilac_Kullanım_Süresi'] = (df['Ilac_Bitis_Tarihi'] -
    ↪df['Ilac_Baslangic_Tarihi']).dt.days

[79]: # Orijinal tarih sütunlarını kaldırma
df.drop(['Dogum_Tarihi', 'Ilac_Baslangic_Tarihi', 'Ilac_Bitis_Tarihi',
    ↪'Yan_Etki_Bildirim_Tarihi'], axis=1, inplace=True)

[80]: from sklearn.preprocessing import StandardScaler

# StandardScaler nesnesi
scaler = StandardScaler()

# Kilo ve Boy'u standardize etme
df[['Kilo', 'Boy']] = scaler.fit_transform(df[['Kilo', 'Boy']])

[81]: df
```

```
[81]:
```

	Kullanici_id	Cinsiyet	Kan Grubu	Kilo	Boy	Alzheimer	Astim	\
0	107	2	8	1.269686	1.015705	0.0	0.0	
1	140	2	0	0.007809	0.394918	0.0	0.0	
2	2	1	8	0.696106	-1.032892	0.0	0.0	
3	83	2	6	0.000000	-0.598341	0.0	8.0	
4	7	1	6	1.040254	-0.163790	18.0	0.0	
...	
2352	9	0	0	-1.770291	-1.405364	0.0	4.0	
2353	101	1	6	0.000000	0.208682	0.0	22.0	
2354	127	1	7	0.524032	1.760649	2.0	8.0	
2355	178	2	0	0.524032	0.581154	1.0	16.0	
2356	174	1	5	-0.106907	0.022446	1.0	0.0	

	Diger	Diyabet	Guatr	...	Yan_Etki_Bildirim_Yili	Dogum_Ayi	\
0	0.0	0.0	18.0	...	2022	3	
1	2.0	16.0	2.0	...	2022	10	
2	16.0	15.0	0.0	...	2022	12	
3	3.0	1.0	0.0	...	2022	6	
4	8.0	9.0	0.0	...	2022	9	
...	
2352	0.0	0.0	4.0	...	2022	1	
2353	0.0	16.0	0.0	...	2022	11	
2354	18.0	8.0	1.0	...	2022	11	
2355	5.0	0.0	0.0	...	2022	1	
2356	19.0	0.0	0.0	...	2022	11	

	Ilac_Baslangic_Ayi	Ilac_Bitis_Ayi	Yan_Etki_Bildirim_Ayi	Dogum_Gunu	\
0	1	3	2	1	
1	1	3	2	12	

2	1	3	2	17
3	1	3	2	17
4	1	3	2	3
...
2352	1	3	2	4
2353	1	3	2	9
2354	1	3	2	29
2355	1	3	2	30
2356	1	3	2	7

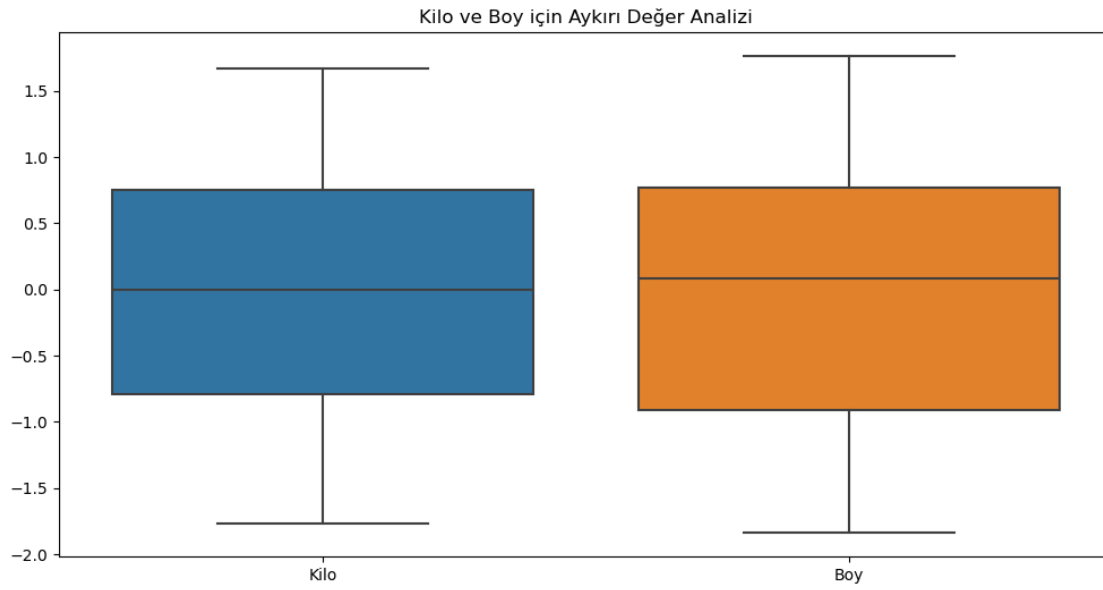
	Ilac_Baslangic_Gunu	Ilac_Bitis_Gunu	Yan_Etki_Bildirim_Gunu	\
0	9	4	19	
1	9	8	3	
2	11	12	4	
3	4	12	8	
4	13	6	12	
...	
2352	13	4	12	
2353	2	5	19	
2354	2	12	3	
2355	2	8	4	
2356	6	6	17	

	Ilac_Kullanım_Süresi
0	54
1	58
2	60
3	67
4	52
...	...
2352	50
2353	62
2354	69
2355	65
2356	59

[2357 rows x 240 columns]

```
[82]: import seaborn as sns
import matplotlib.pyplot as plt

# Boxplot ile Kilo ve Boy aykırı değer analizi
plt.figure(figsize=(12, 6))
sns.boxplot(data=df[['Kilo', 'Boy']])
plt.title('Kilo ve Boy için Aykırı Değer Analizi')
plt.show()
```



```
[83]: # DataFrame'i Excel dosyasına kaydet  
df.to_excel('son hali.xlsx', index=False)
```

```
[ ]:
```