ANALİZLER VE KODLAR

# yan etkilerin dağılımı histogramı

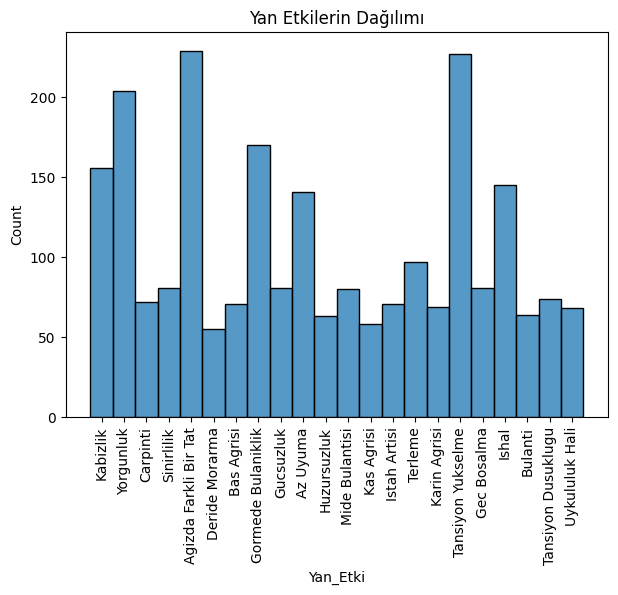
plt.figure(figsize=(7, 5))

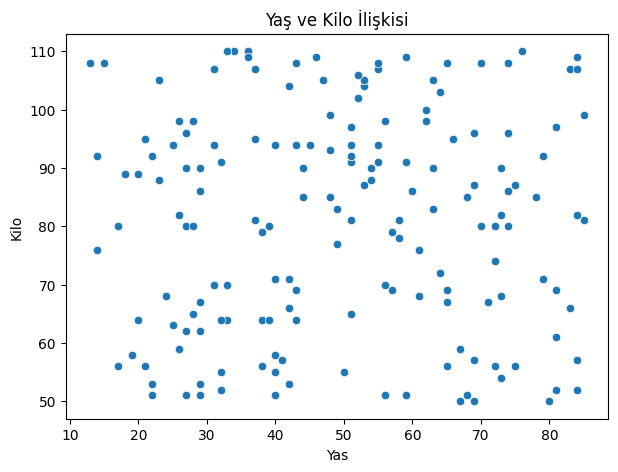
sns.histplot(df['Yan\_Etki'])

plt.xticks(rotation=90)

plt.title('Yan Etkilerin Dağılımı')

plt.show()



# Yaş ve Kilo ilişkisini scatter plot ile göster

df['Dogum\_Tarihi'] = pd.to\_datetime(df['Dogum\_Tarihi'])

df['Yas'] = pd.Timestamp.now().year - df['Dogum\_Tarihi'].dt.year

plt.figure(figsize=(7,5))

sns.scatterplot(x='Yas', y='Kilo', data=df)

plt.title('Yaş ve Kilo İlişkisi')

plt.show()

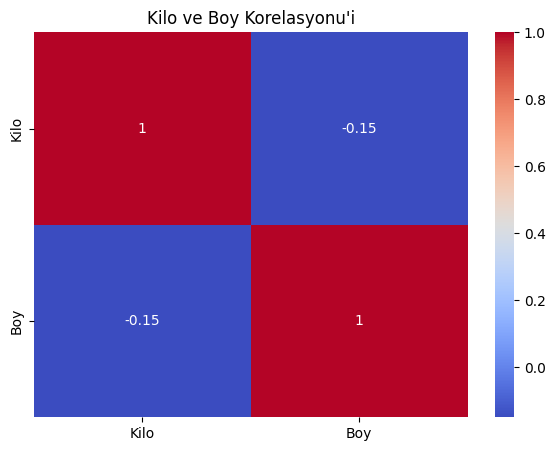
# korelasyon heatmap'i oluştur ( ısı haritası )

plt.figure(figsize=(7, 5))

sns.heatmap(df[['Kilo', 'Boy']].corr(), annot=True, cmap='coolwarm')

plt.title('Kilo ve Boy Korelasyonu\'i')

plt.show()



KODLAR

# EDA

# veri analizi ve incelemeler için kütüphanelerin yüklenmesi

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

# dosya .xlsx formatında .csv formatına çevrilip öyle okunacak.

df = pd.read\_excel("/content/side\_effect\_data 1.xlsx")

df.to\_csv("side\_effect\_data.csv", index=False)

df = pd.read\_csv("side\_effect\_data.csv")

# evet çevrilen dosya hakkında birkaç bilgi

print(df.info())

# verinin temel istatistiklerini görüntüle

print(df.describe(include='all'))

print(df.head())

# eksik verileri kontrol etmek için..

print(df.isnull().sum())

print("--------------------------------------------")

# kategorik değişkenlerin değerleri bunlar da.

print(df['Cinsiyet'].value\_counts()) # cinsiyet

print(df['Yan\_Etki'].value\_counts()) # yan etkiler

# yan etkilerin dağılımı histogramı

plt.figure(figsize=(7, 5))

sns.histplot(df['Yan\_Etki'])

plt.xticks(rotation=90)

plt.title('Yan Etkilerin Dağılımı')

plt.show()

# Yaş ve Kilo ilişkisini scatter plot ile göster

df['Dogum\_Tarihi'] = pd.to\_datetime(df['Dogum\_Tarihi'])

df['Yas'] = pd.Timestamp.now().year - df['Dogum\_Tarihi'].dt.year

plt.figure(figsize=(7,5))

sns.scatterplot(x='Yas', y='Kilo', data=df)

plt.title('Yaş ve Kilo İlişkisi')

plt.show()

# korelasyon heatmap'i oluştur ( ısı haritası )

plt.figure(figsize=(7, 5))

sns.heatmap(df[['Kilo', 'Boy']].corr(), annot=True, cmap='coolwarm')

plt.title('Kilo ve Boy Korelasyonu\'i')

plt.show()

# data pre-processing işlemleri

# importlar

from sklearn.impute import SimpleImputer

# kategorik eksik verileri doldurmak için

kategorilerim = ['Cinsiyet', 'Il', 'Alerjilerim', 'Kronik Hastaliklarim',

'Baba Kronik Hastaliklari', 'Anne Kronik Hastaliklari',

'Kiz Kardes Kronik Hastaliklari', 'Erkek Kardes Kronik Hastaliklari',

'Kan Grubu']

# en sık tekrar eden değerle doldurulması..

imputer\_cat = SimpleImputer(strategy='most\_frequent')

df[kategorilerim] = imputer\_cat.fit\_transform(df[kategorilerim])

# sayısal eksik verilerin doldurulması

sayısallar = ['Kilo', 'Boy']

# ortalamalarla doldurulur.

imputer\_num = SimpleImputer(strategy='mean')

df[sayısallar] = imputer\_num.fit\_transform(df[sayısallar])

# eksik verileri tekrar kontrol etmek için..

print(df.isnull().sum()) # tamamen sıfırlanmıştır.

# genel bilgi

print(df.info())

#istatistiksel bilgi

print(df.describe(include='all'))