**SEABORN**

**01-2.05.2023**

[https://seaborn.pydata.org/](https://seaborn.pydata.org/" \t "/Users/merthan/Desktop/IT_Course/Data Visualization/x/_blank)

#seaborn matplotlibe gore daha cok imkan veriyor gorsellestirme acisindan

|  |
| --- |
| **import pandas as pd**  **import numpy as np**  **import matplotlib as mpl**  **import matplotlib.pyplot as plt**  **import seaborn as sns**  **import warnings**  **warnings.filterwarnings("ignore")** |

**Seaborn icindeki datasetlerini cagirmak:**

**sns.get\_dataset\_names()**

#seaborn icindeki datasetleri getirdi.

**Seaborn icindeki belirli bir datasetini okutmak:**

**tips= sns.load\_dataset("tips")**

#seaborn tips datasetini okudum.

|  |
| --- |
| **Datasetine asina olmak icin sirayla bak:**   1. **tips.head()** 2. **tips.tail()** 3. **tips.info()** 4. **tips.describe()** 5. **tips.sample(10)**-#rastgele-randomize 10 satir getirir. Birkac kere calistirip bakilabilir. 6. **tips["day"].unique()**-#kategorik datalar icin unique olanlara mutlaka bak |

## **Scatter Plot**

|  |
| --- |
| **sns.scatterplot(data=datasetinin adi, x, y)** |

## Dagilim grafikleri, farkli ozelliklerin birbiriyle olan iliskisini (korelasyonunu) goruntulemek icin kullanilir.

**sns.scatterplot(data=tips, x="total\_bill", y="tip")**

**plt.show()**

**sns.scatterplot(x="total\_bill", y="tip", data=tips)**

**plt.show()**

#ikisi de ayni sonucu verir. x,y, data nin sirasinin bir onemi yok. positional argument olmadigi surece

**sns.regplot(x="total\_bill", y="tip", data=tips);**

#bu da bir cizgi ile birlikte verir dagilimi.

#bu bahsettigimiz cizgi regresyon cizgisi.

#altindakilerle ustundekilerin, cizgiye mesafeleri toplami 0. alt eksi ust arti kabul edilir.

**Figsize**

**plt.figure(figsize = (10, 5))** #figure’un buyuklugunu ayarladigimiz kisim.

**sns.scatterplot(x = tips["total\_bill"], y = tips["tip"])**

#tips datasetinden total\_billi al x-axis’te goster, tipi al y axiste goster.

**plt.show()**

**Hue**

# hue targetta kullaniyoruz. renklendiren parametremiz.

#sns.scatterplot shift tab tab diyince zaten icine hue verebildigimizi goruyoruz.

#ozellikle **kategorik veri icin** kullaniyoruz.

#size gibi numeric olan kategoric veri icin de kullanilabilir.

**plt.figure(figsize = (10, 5))**

**sns.scatterplot(x = "total\_bill", y = "tip", data = tips, hue = "smoker" )**

**plt.show()**

#hue=”smoker” diyerek ayni grafikte smokerlara gore tip verilmesini renklendirmis olduk.

#smokerlarin daha cok daha az tip vermesiyle ilgili bir insight cikaramiyoruz.

**plt.figure(figsize = (10, 5))**

**sns.scatterplot(x = "total\_bill", y = "tip", data = tips, hue = "sex" )**

**plt.show()**

#burda cinsiyete gore renklendirdik.

#insight:kadinlar daha az tip veriyor gibi gorunuyor.

**plt.figure(figsize = (10, 5))**

**sns.scatterplot(x = "total\_bill", y = "tip", data = tips, hue = "day" )**

**plt.show()**

#burda gune gore renklendirmis olduk. Bu da fazla insight saglamadi.

**Hue and Style**

**plt.figure(figsize = (12,8))**

**plt.subplot(121)**

**sns.scatterplot(x="total\_bill", y="tip", style="sex", data=tips)**

**plt.subplot(122)**

**sns.scatterplot(x="total\_bill", y="tip", hue="smoker", data=tips)**

**plt.show()**

#yani burda hue bir gruplama yapiyor. renge gore ayristiriyor. turuncu mavi

#style ise sekle gore ayristirarak grupluyor. Biri dot biri yildiz gibi.

**plt.figure(figsize = (10, 5))**

**sns.scatterplot(data = tips, x = "total\_bill", y = "tip", style = "day", hue = "sex");**

#bu da mesela, cinsiyete gore renkli bir gruplama yapiyor, gunlere gore ise sekillerini farklilastiriyor.

**fig, ax = plt.subplots(1, 2, figsize = (10, 5))**

**sns.scatterplot(x = "total\_bill", y = "tip", data = tips, style = "sex", ax = ax[0])**

**sns.scatterplot(x = "total\_bill", y = "tip", data = tips, hue = "smoker", ax = ax[1])**

**plt.show()**

**Palette**

**plt.figure(figsize=(10,5))**

**sns.scatterplot(data=tips, x="total\_bill", y="tip", hue="day", palette="plasma")**

#hue'ya cinsiyet verince cinsiyete gore renklendirecek

#palette de renk opsiyonlarini veriyor bana daha once mavi turuncu deafult veriyordu

#icinde plasma, rainbow, cividis gibi opsiyonlar var palettenin

**Density**

**plt.figure(figsize=(10,5))**

**sns.scatterplot(data=tips, x="total\_bill", y="tip", alpha=0.5);**

#alpha da koyulugunu ayarliyor.

**little little into the middle**

**plt.figure(figsize=(10,5))**

**sns.scatterplot(data=tips, x="total\_bill", y="tip", alpha=0.6, hue="day", style= "sex", palette="plasma");**

#scatter plot genelde distribution icin kullanilir, classlama icin genelde,

#ogreniclerin notlari, kan degerleri, hava sicakliklari sayidir.

**S**

plt.figure(figsize=(10,5))

sns.scatterplot(data=tips, x="total\_bill", y="tip", alpha=0.6, hue="day", style= "sex", palette="plasma", s=400);

#bu da s parametresi sekilleri buyutuyor. Dot seklinde gosteriyorsa mesela buyuk dot yapiyor.

**Size**

**plt.figure(figsize=(10,5))**

**sns.scatterplot(data=tips, x="total\_bill", y="tip", alpha=0.5, hue="smoker", style= "day", size="sex");**

#size=”sex” diyince, kucuk yuvarlagi kadin, buyuk yuvarligi erkek yapti. Isaretin birini kucuk birini buyuk yapiyor verdigin kategorik veri tipine gore.

#ozetle, **hue, style ve size** ile cesitli gruplamalar yapiyor.

(#the relationship between ``x`` and ``y`` can be shown for different subsets of the data using the ``hue``, ``size``, and ``style`` parameters.)

**Distribution Plot**

1. **sns.displot(x="total\_bill", data=tips);**

#burda y ekseninde count veriyor.

#displotlarda boyle x eksenine bir deger veriyorum, **y count** oluyor.

#mesela bu grafikte 10 ile 13 arasinda bahsis verme frekansi 45 diyor. Yani 45 kisi bu aralikta bahsis vermis.

#burda bir dagilim egrisi goruyorum.

#right-skeweness denebilir buna.

#herbir bara bins diyoruz.

1. **sns.displot(x="total\_bill", data=tips, kind='kde')**

#kind='kde' diyince dagilim egrisi ciziyor.

#bu da right-skewed.

1. **sns.displot(x="total\_bill", data=tips, kde=True)**

#bu da hem barlari hem de kde egrisini gosteriyor.

#yogunlunugun 10-20 arasinda oldugunu anlayabiliyorum.

1. **sns.displot(x="total\_bill", data=tips, bins=100)**

#100 bara cikardik bins=100 diyerek, 50 desek 50 barla yapar.

#kde yogunlugu gosteriyor

#bins artirinca barlarin sayisi artmis oluyor.

#bir nevi zoom yapiyoruz dataya.

#y ekseninde count gosterdigi icin, aralik sayisi arttikca,

#haliyle o araliktaki frekans yani gozlem sayisi dusecek. o yuzden binsi artirdikca biz y ekseninin araligi daralir. Barin genisligi de daralir.

#ne kadar azsa bins, bar basina o kadar cok gozlem duser.

1. **sns.displot(x="total\_bill", data=tips, bins=100, kde=True)**

#burda da kde yogunlugu gosteriyor. count yetmez yogunluk icin.

#kde=True dedigim icin barlarla birlikte kde yogunluk cizgisini de gosterdi

#yogunluga bakinca da anliyoruz ki 10-20 arasinda tipler yogunlasiyor.

#scatterplot'da y eksenine tip diye verdim

#histplotta ise y'yi dogrudan count olarak hesaplatiyor

1. **Subplots**

**fig, ax = plt.subplots(1,2, figsize = (10,5))**

**sns.scatterplot(data = tips, x = "total\_bill", y = "tip", s = 300, hue = "sex", ax = ax[0])**

**sns.histplot(data = tips, x = "total\_bill", ax = ax[1], kde =True)**

#bunlar ayni seyi cizdiriyor aslinda. biri y ekseninde tip, digeri count gosteriyor.

#yanyana iki plot olacak sekilde cizdirdigim bu iki plotun ikisinden de 10-20 arasindaki yogunslasma anlasiliyor

1. **FacetGrid ve map**

**g=sns.FacetGrid(data=tips, col="sex", hue="smoker")**

**g.map(sns.scatterplot, "total\_bill", "tip", alpha=.5)**

**g.add\_legend();**

#gorsellerin dagilimini gostermis oluyor FacetGrid

#yogunlugun erkeklerde daha fazla oldugunu gosteriyor. erkekler daha cok tip veriyormus.

#col=sex dedigim icin cinsiyete gore iki plot yapti.

#hue=smoker dedigim icin smokerlara gore ayirdi.

1. **shade=True**

**g=sns.FacetGrid(data=tips, col="sex", hue="smoker")**

**g.map(sns.scatterplot, "total\_bill", "tip", alpha=.5, shade=True)**

**g.add\_legend();**

#shade=True deyince, golgeli sekilde bir grafik cizdirdi

#yogunlugu haritalardaki gibi daha koyu gosteriyor.

#isi haritasi goruntusu gibi veriyor

1. **Aspect**

**sns.displot(data=tips, x= "total\_bill", kind="hist", kde=True, aspect=1)**

#aspect yatay duzlemde genisletiyor

#aspect'in defaultu 1. 10 yaparsam yatay duzlemde genisletmis oluyor. diger her sey ayni kalir.

**Hist plot**

**plt.figure(figsize= (10,5))**

**sns.histplot(data= tips, x="total\_bill", bins=50, kde=True);**

#bu da dagilimi gosteren bir grafik.

**Doppler Effect**

#rugplot ile histogram plotun ayni anda olmasi

#doppler effect deniyor

**fig, ax = plt.subplots(1, 2, figsize = (10, 5))**

**sns.rugplot(data=tips, x="total\_bill", ax=ax[0])**

**sns.histplot(data=tips, x="total\_bill", ax=ax[1], kde=True)**

**fig, ax = plt.subplots(1, 2, figsize = (10, 5))**

**sns.rugplot(data = tips, x = "total\_bill", height = 0.1, ax = ax[0])**

**sns.histplot(data = tips, x = "total\_bill", bins = 50, kde = True, ax = ax[1]);**

#rugplotun icine girip height dersem, rugplot barlarinin yuksekligini degistirebilirim.

#rugplotun en guzel kiyaslandigi grafik histogramdir.

#rugplot, aslinda histogramin izdusumu gibidir.

**sns.histplot(data=tips, x="tip", kde=True)**

**sns.set(style="dark")**

#kde ile yogunluk egrisini de gosteren, x ekseninde tip olan, y’de de count olan bir histogram cizdi.

**KDE**

**sns.kdeplot(data = penguins, x = "body\_mass\_g");**

#penguins datasetinin, e ekseninde body\_mass\_g olan kde plotunu cizdiriyor.

#y ekseninde density olacak kde cizdirdigim icin

**sns.kdeplot(data = penguins, x = "body\_mass\_g", hue = "sex");**

**sns.kdeplot(data = penguins, x = "body\_mass\_g", hue = "sex", shade=True);**

#KDE ile birbirinden ayrik plotlar icin daha guvenli tahminlerde bulunabilirim

#birbirinin icine giren kisimlar icin modelim hata yapabilirim.

**2D KDE graph (Izohips benzeri)**

**sns.kdeplot(data = tips, x = "total\_bill", y = "tip");**

#hem x hem y ekseni verirsem, 2-D KDE plot cizdirir.

**sns.kdeplot(data = tips, x = "total\_bill", y = "tip", color = "red", shade=True);**

#yogunlugu renk koyulugu ile gosterir

**fig, ax = plt.subplots(1, 2, figsize=(10, 5))**

**sns.kdeplot(data=tips, x="total\_bill", y="tip", color="red", shade=True, ax=ax[1])**

**sns.scatterplot(data=tips, x="total\_bill", y="tip", color="red", ax=ax[0])**

**plt.show()**

#ax=0'a scatterplot cizdi.ilk plot. ax=1’e ikinciye de golgeli kdeplot

#ikisinin de x=totalbill, y tip, yogunlugu gosteriyor ikisi de.

**Kde, hist, dist in same plot**

**sns.displot(data = penguins,**

**x = "body\_mass\_g",**

**bins = 20,**

**rug = True,**

**kde = True);**

#displot diyerek displot cizdirdik, rug=True, kde=True ile onlari da ayni plota ekledik

#kde, hist, dist 3 tane distribution plotu ayni grafikta gormus olduk.

**2 rows 2 columns subplots**

**fig, ax = plt.subplots(2, 2, figsize=(10, 5))**

**sns.kdeplot(data=tips, x="total\_bill", y="tip", color="purple", shade=True, ax=ax[0, 0])**

**sns.scatterplot(data=tips, x="total\_bill", y="tip", color="red", ax=ax[0, 1])**

**sns.histplot(data=tips, x="total\_bill", y="tip", color="orange", ax=ax[1, 0])**

**sns.rugplot(data=tips, x="total\_bill", height=0.5, ax=ax[1, 1])**

**plt.show()**

#kde, scatter, histplot, rugplot as 4 different plots in same figure

**CARS DATASET**

**sns.displot(data=cars, x="weight", hue="origin");**

#hue origin diyerek arabalari origin yerlerine gore renklendirdik.

#3000 kg'dan daha agir olanlarin hepsi ABD'de mesela

**sns.displot(data = cars[cars.cylinders.isin([4, 6, 8])], x = "weight",**

**col = "origin",**

**row = "cylinders",**

**hue = "cylinders");**

#bu bana 3x3 9 tane plot verdi.

#bu kodla satirlardaki basliklari elde ediyorum ve legend koydurmus oluyorum

#**sns.displot(data = cars[cars.cylinders.isin([4, 6, 8])]** koduyla

#cars datasetinden, cylinders column'unun 4,6,8 degerlerine gore grafik cizdirmis olduk

#isin kodu, bir data setinin herhangi columnundan bazi degerlere gore grafik olusturmak icin kullanilir.