

MATPLOTLIB

Verileri görselleştirip, 2 ve 3 boyutlu çizimler yapmamızı sağlar.
Şimdi TÜFE verisini grafik üzerinden inceleyelim.

```
In [ ]: import matplotlib.pyplot as plt  
import pandas as pd
```

```
In [ ]: data = pd.read_excel("tufe.xls")
```

```
In [ ]: data.index=[ "2003", "2004", "2005", "2006", "2007", "2008",  
"2009", "2010", "2011", "2012", "2013", "2014",  
"2015", "2016", "2017", "2018", "2019", "2020",  
"2021", "2022", "2023"]  
data.drop("Yıl",axis=1,inplace=True)
```

```
In [ ]: #data["Ocak"] = data["Ocak"].astype("float")  
#data["Şubat"] = data["Şubat"].astype("float")
```

```
In [ ]: data.head()
```

```
Out[ ]:
```

	Ocak	Şubat	Mart	Nisan	Mayıs	Haziran	Temmuz	Ağustos	Eylül	Ekim	Kasın
2003	94.77	96.23	98.12	99.09	100.04	100.12	99.93	100.09	101.44	102.38	103.66
2004	104.81	105.35	106.36	106.89	107.35	107.21	107.72	108.54	109.57	112.03	113.50
2005	114.49	114.51	114.81	115.63	116.69	116.81	116.14	117.13	118.33	120.45	122.14
2006	123.57	123.84	124.18	125.84	128.20	128.63	129.72	129.15	130.81	132.47	134.18
2007	135.84	136.42	137.67	139.33	140.03	139.69	138.67	138.70	140.13	142.67	145.41

Temel Görselleştirme ve Genel Komutlar

plt.plot() grafik oluşturmanın temel komutudur. Grafiği tanımladıktan sonra, özelliklerini ekleriz.

plt.figure(figsize=()), grafiğin boyutunu belirler. x eni, y ise boyudur.

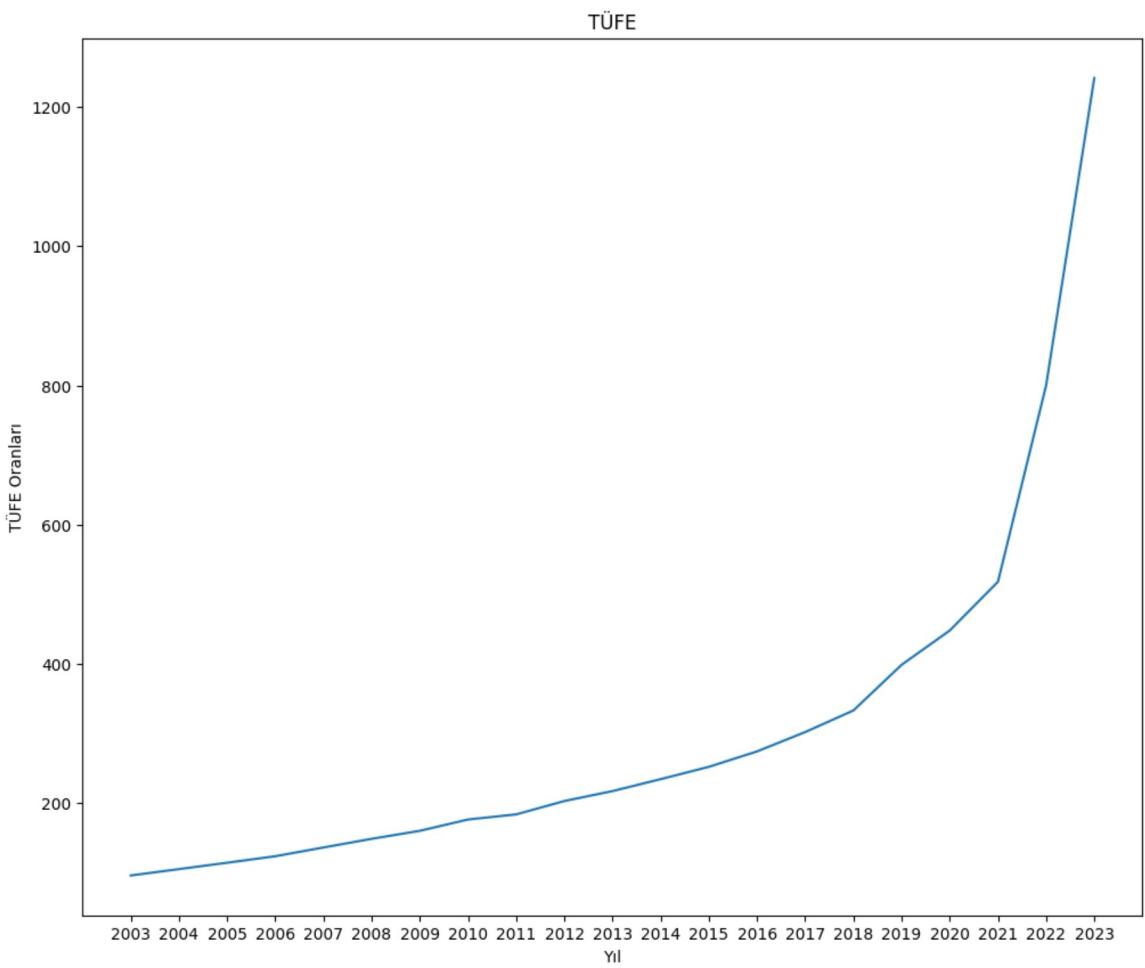
plt.xlabel(), x eksesine ait başlık bilgisi

plt.ylabel(), y eksesine ait başlık bilgisi

plt.title(), grafiğin başlık bilgisidir.

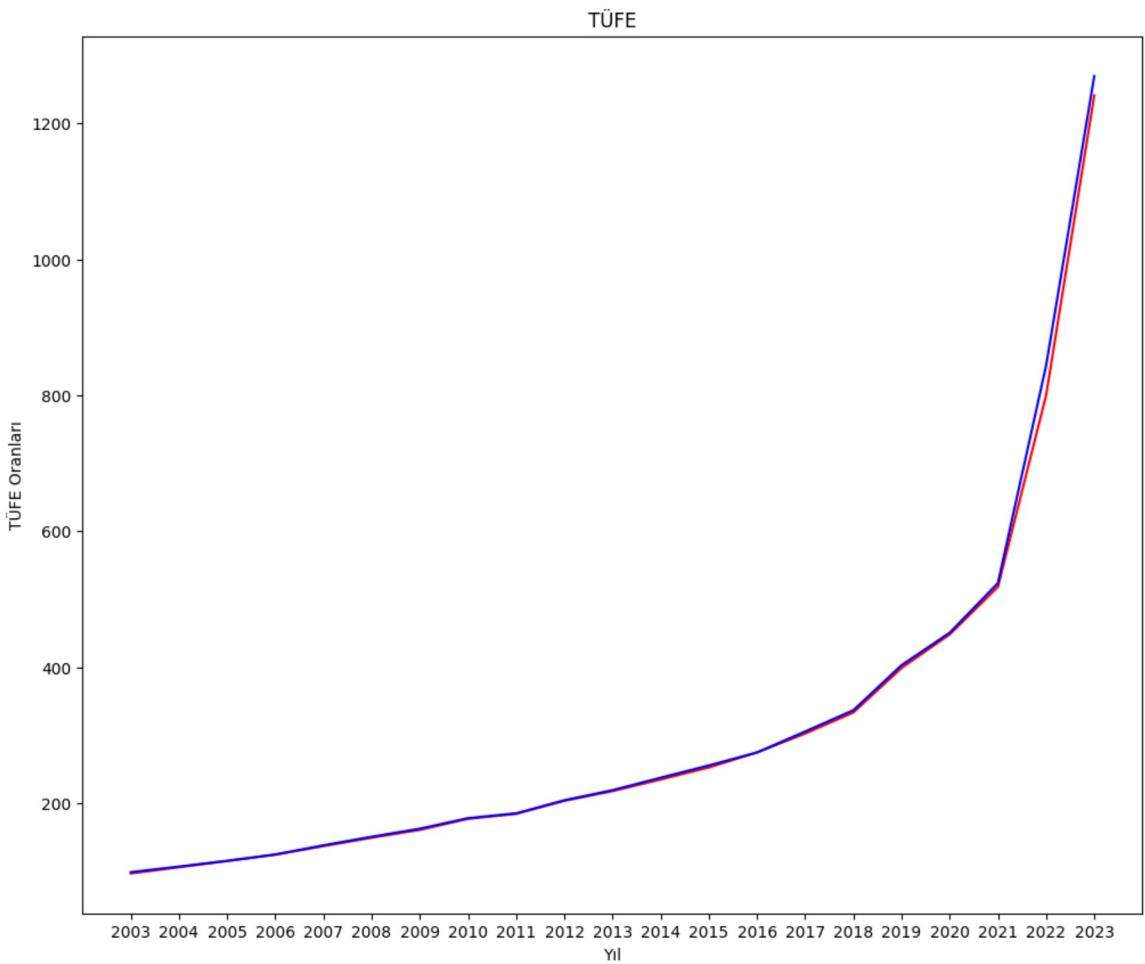
plt.show(), grafiğin çizilmesini ve ekranda gözükmescini sağlar.

```
In [ ]: plt.figure(figsize=(12,10))  
plt.plot(data.index, data["Şubat"])  
plt.title("TÜFE")  
plt.xlabel("Yıl")  
plt.ylabel("TÜFE Oranları")  
plt.show()
```



İki farklı veriyi tek grafik üzerinde gösterelim. Burada dikkat etmemiz gereken konu, iki verinin de x veya y eksenlerinin ortak olmasıdır.

```
In [ ]: plt.figure(figsize=(12,10))
plt.plot(data.index, data["Şubat"], color="red")
plt.plot(data.index, data["Mart"], color="blue")
plt.title("TÜFE")
plt.xlabel("Yıl")
plt.ylabel("TÜFE Oranları")
plt.show()
```

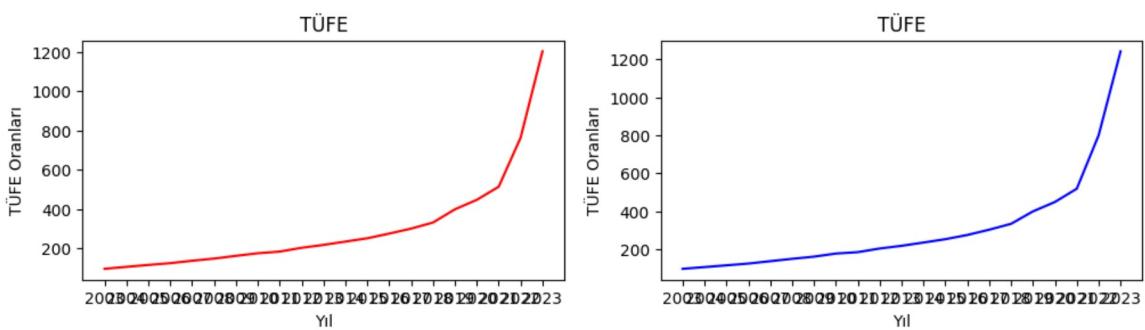


subplot(x,y,z) metodu birden fazla grafiği aynı anda çizmemizi sağlar. x satır, y sütun ve z ise kaçinci grafik olduğunu ifade eder.

```
In [ ]: plt.figure(figsize=(12,6))

plt.subplot(2,2,1)
plt.plot(data.index, data["Ocak"], color="red")
plt.title("TÜFE")
plt.xlabel("Yıl")
plt.ylabel("TÜFE Oranları")

plt.subplot(2,2,2)
plt.plot(data.index, data["Şubat"], color="blue")
plt.title("TÜFE")
plt.xlabel("Yıl")
plt.ylabel("TÜFE Oranları")
plt.show()
```



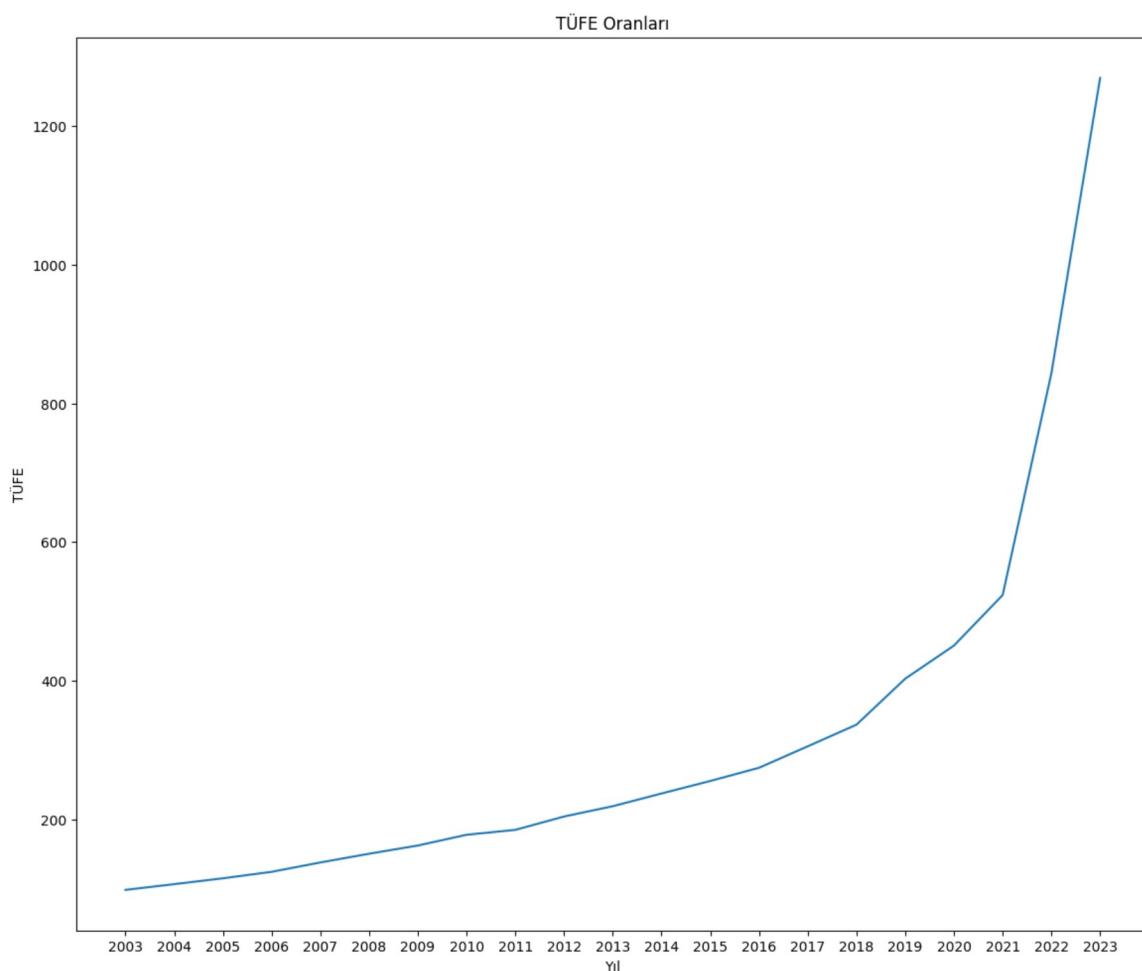
Figür Tanımlamak ve Çizmek

`plt.figure()`, komutu ile de grafik çizebiliriz. Figure sayesinde tablo içerisinde çizimler yapabiliriz.

`.add_axes()`, koordinatları belirlememizi sağlar.

İlk değer soldan ne kadar içerde başlayacağı, ikinci değer alttan ne kadar yüksekte olacağı, üçüncü değer ne kadar uzunlukta olması gerektiği, dördüncü değer ise ne kadar yükseklikte olması gerektiğini belirtmektedir. 0 ile 1 arasında değer almalıdır.

```
In [ ]: fig = plt.figure(figsize =(12,10))
axes = fig.add_axes([0.1,0.1,0.9,0.9])
axes.plot(data.index, data["Mart"])
axes.set_xlabel("Yıl")
axes.set_ylabel("TÜFE")
axes.set_title("TÜFE Oranları")
plt.show()
```



İç İçe Grafik Çizme

Farklı verileri karşılaştırmak ve daha iyi incelemek için iç içe grafikler faydalı olmaktadır. Bunun için 2.axes'i oluşturmamız gereklidir. Burada dikkat edilmesi gereken nokta grafiklerimizin koordinatlarıdır.

```
In [ ]: fig = plt.figure(figsize =(12,10))
```

```
axes1 = fig.add_axes([0.1,0.1,0.9,0.9])
axes2 = fig.add_axes([0.3,0.4,0.5,0.4])

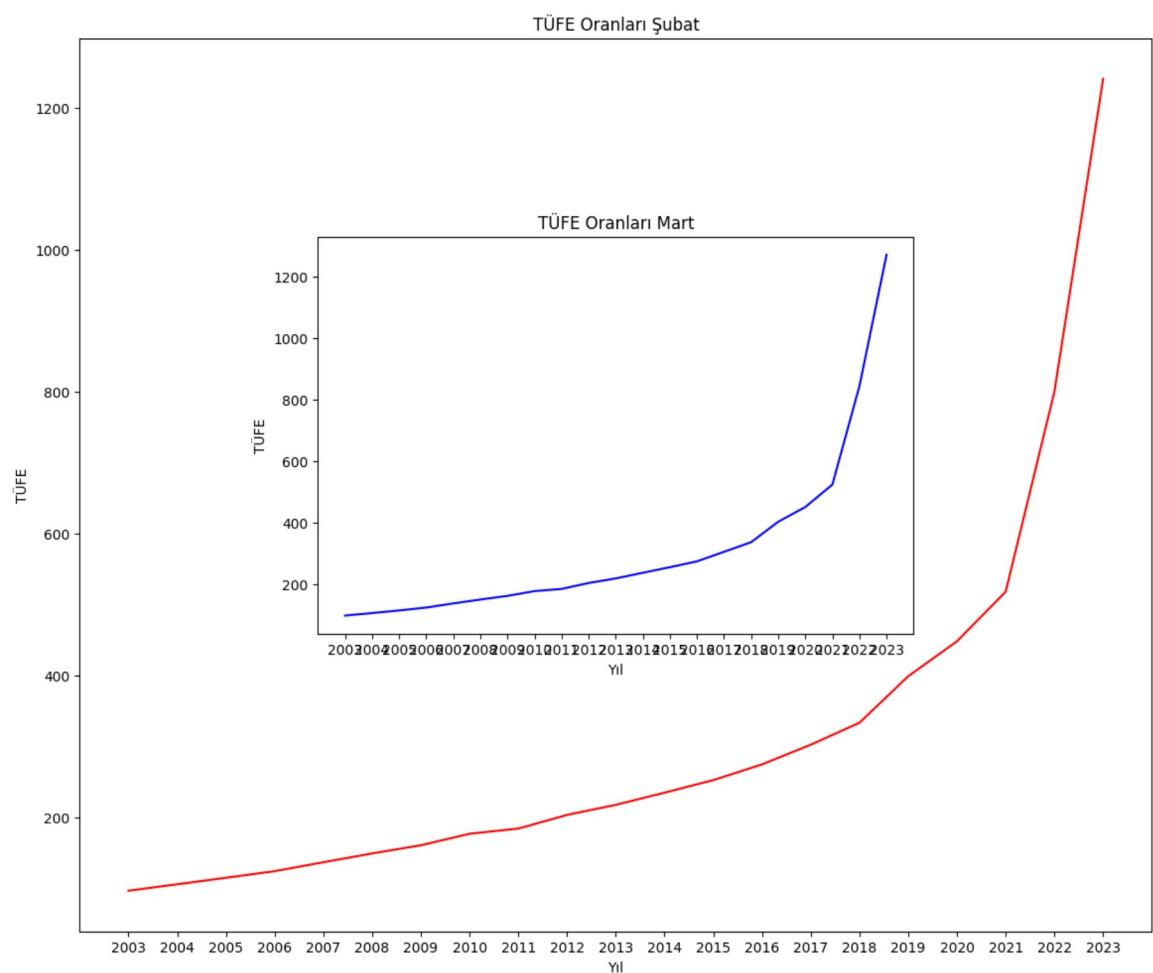
axes1.plot(data.index, data["Şubat"],color="red")
axes2.plot(data.index, data["Mart"],color="blue")

axes1.set_xlabel("Yıl")
axes2.set_xlabel("Yıl")

axes1.set_ylabel("TÜFE")
axes2.set_ylabel("TÜFE")

axes1.set_title("TÜFE Oranları Şubat")
axes2.set_title("TÜFE Oranları Mart")

plt.show()
```



Grafiklerin Özelleştirilmesi

Grafiği çizdıktan sonra aşağıda bulunan parametreler ile farklı özellikler tanımlayabiliriz.

color, grafiğin rengini belirler. Farklı renk paletleri yazılabilir.

linewidth, grafik çizgisinin kalınlığını belirler.

linestyle, çizginin stilini belirler.

marker, her kırılım noktasına belirlediğimiz şekli koyar.

markersize, marker boyutunu belirler.

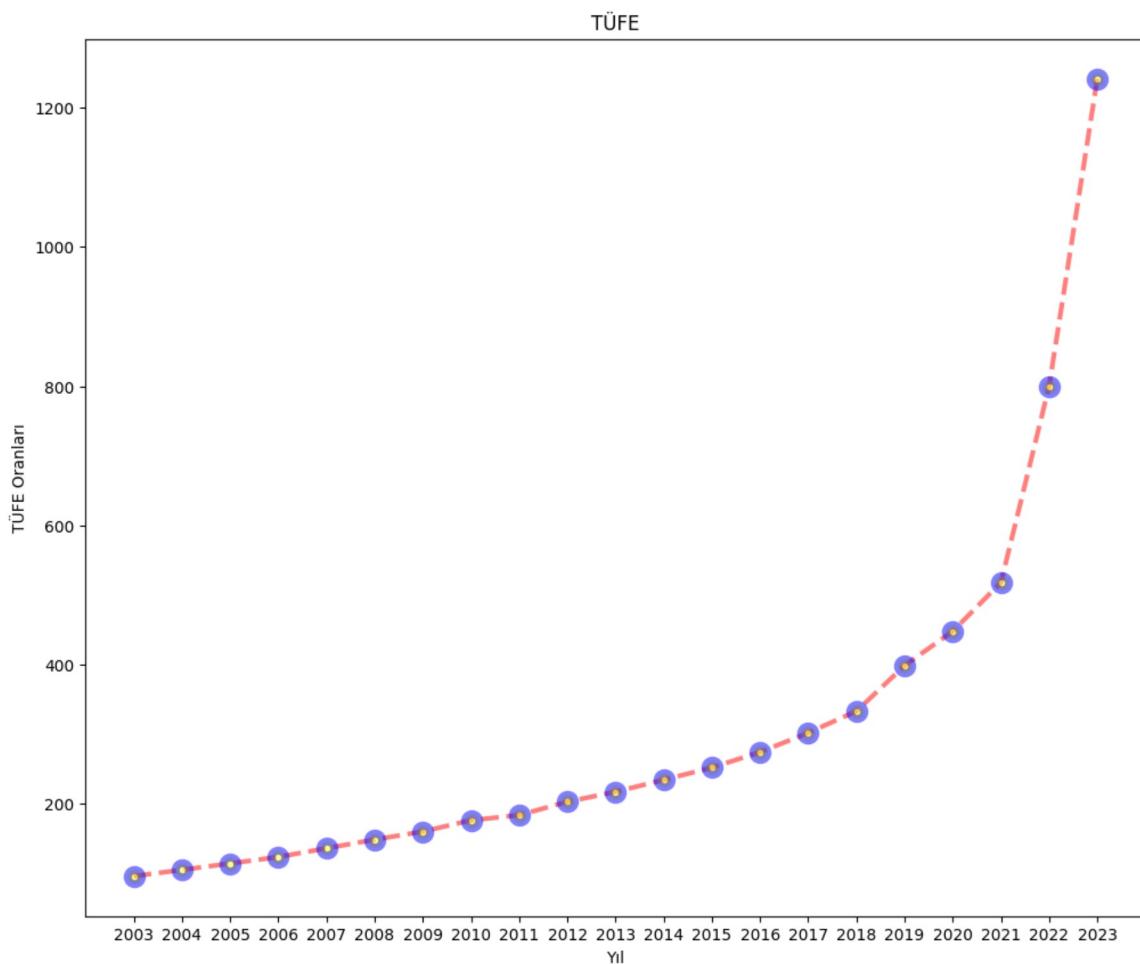
markerfacecolor, rengini değiştirir.

markeredgecolor, çizgisini kalınlaştırır.

markeredgewidth, çizgisinin dış rengini değiştirir.

alpha, grafiğin saydamlığını değiştirir. 0'a yaklaştıkça saydamlaşır. 1'e yaklaştıkça belirginleşir.

```
In [ ]: plt.figure(figsize=(12,10))
plt.plot(data.index, data["Şubat"], color="red", linewidth=3,
          linestyle="--", marker="o", markersize=9,
          markerfacecolor="yellow", markeredgewidth=5,
          markeredgecolor="blue", alpha=0.5)
plt.title("TÜFE")
plt.xlabel("Yıl")
plt.ylabel("TÜFE Oranları")
plt.show()
```



Önceki çalışmada yaptığımız gibi, iç içe grafiklerin her birinde bu özellikleri kullanabiliriz.

Grafik Üzerine Yazı Eklenmesi

label ve **legend**, grafiklerin hangi değişkenlere ait olduğunu gösterir. İkisini aynı anda kullanmalısınız. **legend(loc=2)** şeklinde değer alabilir. Değer 0-10 arasında olmalıdır ve 0 en iyi yeri ifade eder.

bbox_to_anchor=(0.5,0.,0.5,0.5), direkt ledengin konumunu belirler.

fontsize, Font boyut ayarıdır. Aldığı değerler, 'xx-small', 'x-small', 'small', 'medium', 'large', 'x-large', 'xx-large' dır.

numpoints, legendların sayısını arttırır.

markerscale, legendların boyutunu değiştirmemizi sağlar.

markerfirst, Legendi başa, göstergesini sona alır. True ya da False değer alır.

shadow, legendin etrafındaki çizgilere gölgeler verir. True ya da False ile kullanılır.

framealpha, çizgilerinin saydamlığını ayarlar. 0-1 arasında değer alır. 1'e yaklaşıkça saydamlık artar. 0'a yaklaşıkça azalır.

facecolor, arka planına renk vermeye yarar.

edgecolor, çizgisinin renklendirmeyi sağlar.

mode, legend kutucuğunun uzunluğunu belirler. **expand** ya da **None** değerini alır.

expand değerini verdığımızda grafik uzunluğunda uzar.

title, başlık vermeye yarar.

title_fontsize, aşlığının boyutunu belirler. Title ile birlikte kullanılır. Integer değerler alır.

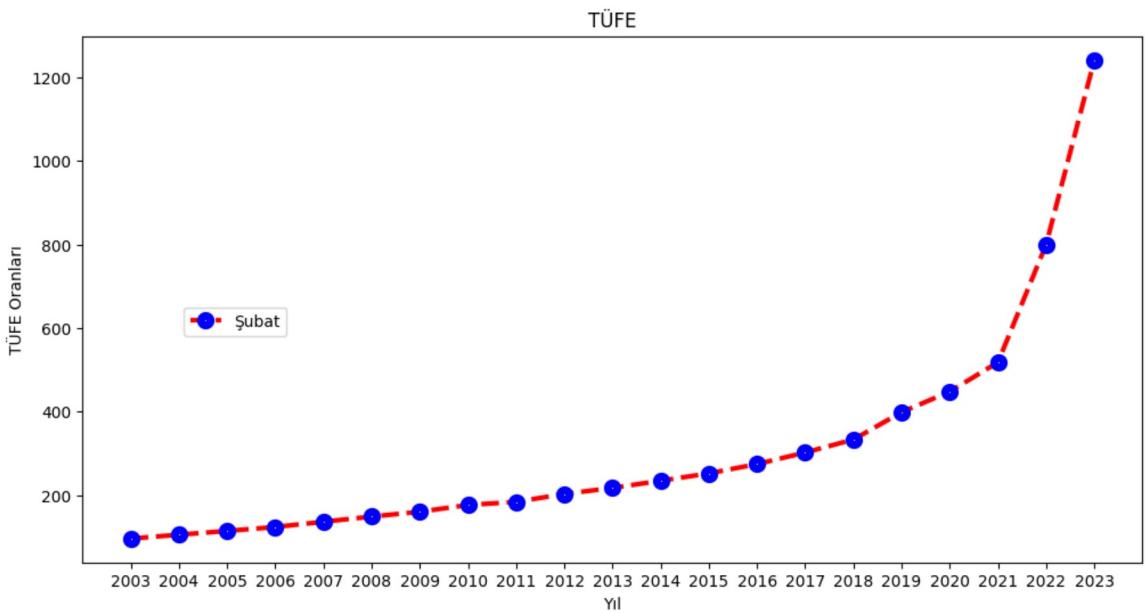
borderpad, legend çerçevesinin boyutunu büyütür. Float değerler alır.

handlelength, kollarının uzunluğunu belirler. Float değerler alır.

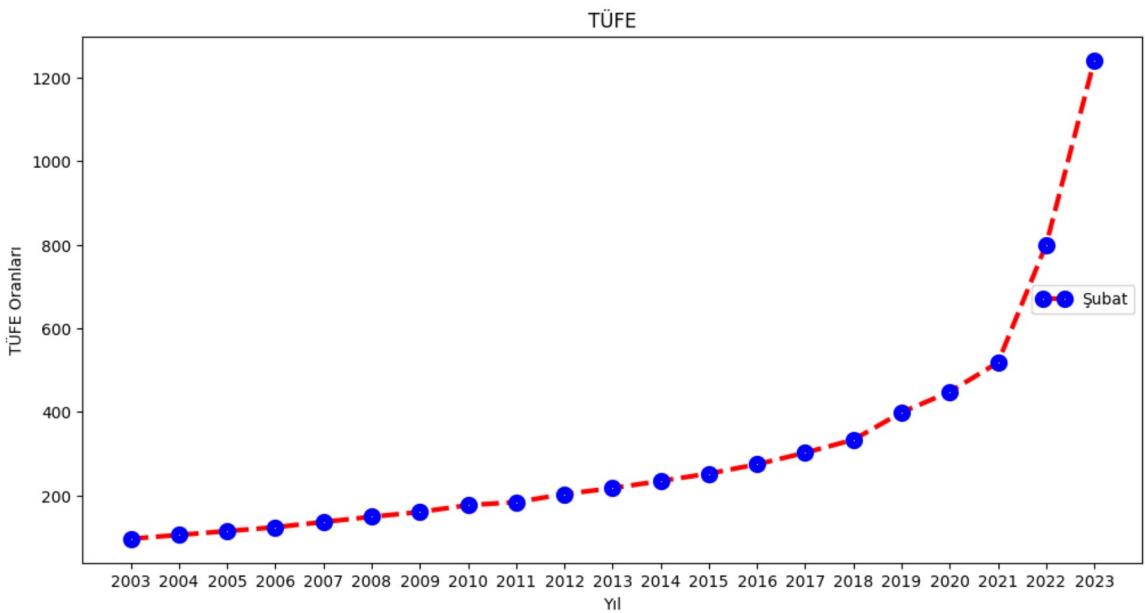
handletextpad, legend noktası ile yazısı arasındaki boşluğu ayarlamamıza yarar.

borderaxespad, legend ile grafik çizgileri arasındaki boşlukları ayarlamamıza yarar. Float değerler alır.

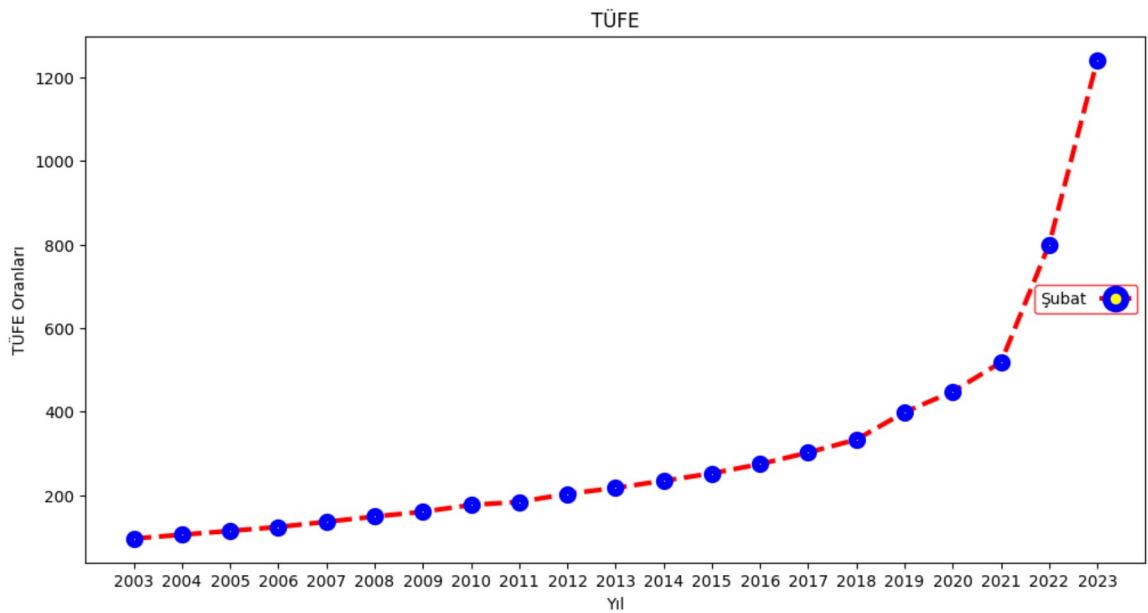
```
In [ ]: plt.figure(figsize=(12,6))
plt.plot(data.index, data["Şubat"], color="red", linewidth=3,
          linestyle="--", marker="o", markerfacecolor="yellow",
          markeredgewidth=5, markeredgecolor="blue")
plt.title("TÜFE")
plt.xlabel("Yıl")
plt.ylabel("TÜFE Oranları")
plt.legend(["Şubat"], bbox_to_anchor=(0.1, 0., 0.1, 0.5))
plt.show()
```



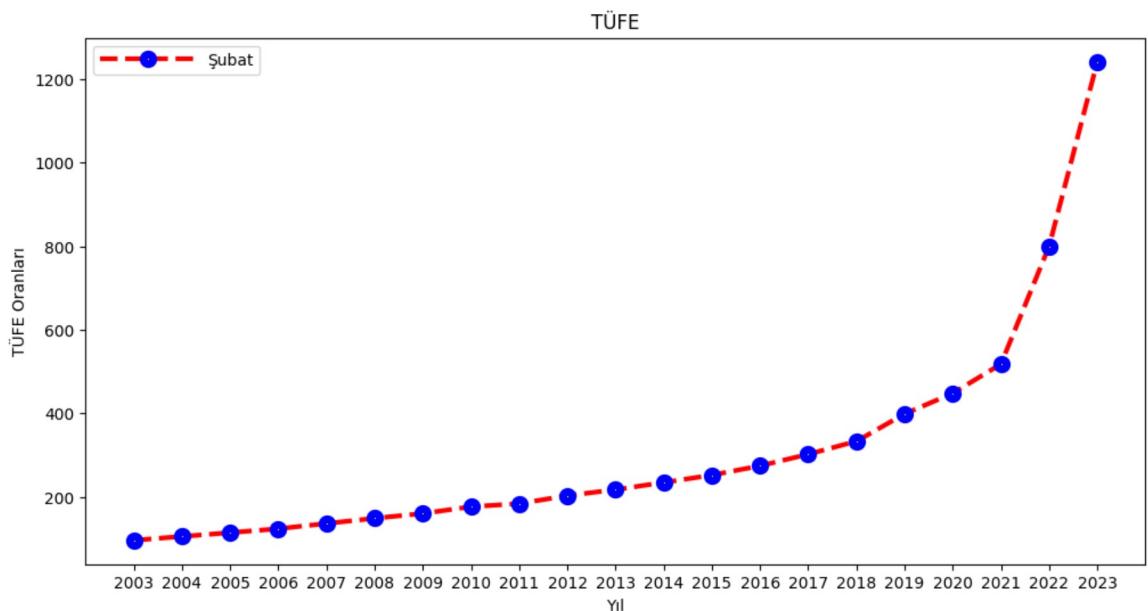
```
In [ ]: plt.figure(figsize=(12,6))
plt.plot(data.index, data["Şubat"], color="red", linewidth=3,
          linestyle="--", marker="o", markerfacecolor="yellow",
          markeredgewidth=5, markeredgecolor="blue")
plt.title("TÜFE")
plt.xlabel("Yıl")
plt.ylabel("TÜFE Oranları")
plt.legend(["Şubat"], loc=5, fontsize="medium", numpoints=2)
plt.show()
```



```
In [ ]: plt.figure(figsize=(12,6))
plt.plot(data.index, data["Şubat"], color="red", linewidth=3,
          linestyle="--", marker="o", markerfacecolor="yellow",
          markeredgewidth=5, markeredgecolor="blue")
plt.title("TÜFE")
plt.xlabel("Yıl")
plt.ylabel("TÜFE Oranları")
plt.legend(["Şubat"], loc=5, fontsize="medium",
           markerscale=2, markerfirst=False, edgecolor="r")
plt.show()
```



```
In [ ]: plt.figure(figsize=(12,6))
plt.plot(data.index, data["Şubat"], color="red", linewidth=3,
          linestyle="--", marker="o", markerfacecolor="yellow",
          markeredgewidth=5, markeredgecolor="blue")
plt.title("TÜFE")
plt.xlabel("Yıl")
plt.ylabel("TÜFE Oranları")
plt.legend(["Şubat"], loc=2, fontsize="medium",
           title_fontsize=16, handlelength=6)
plt.show()
```

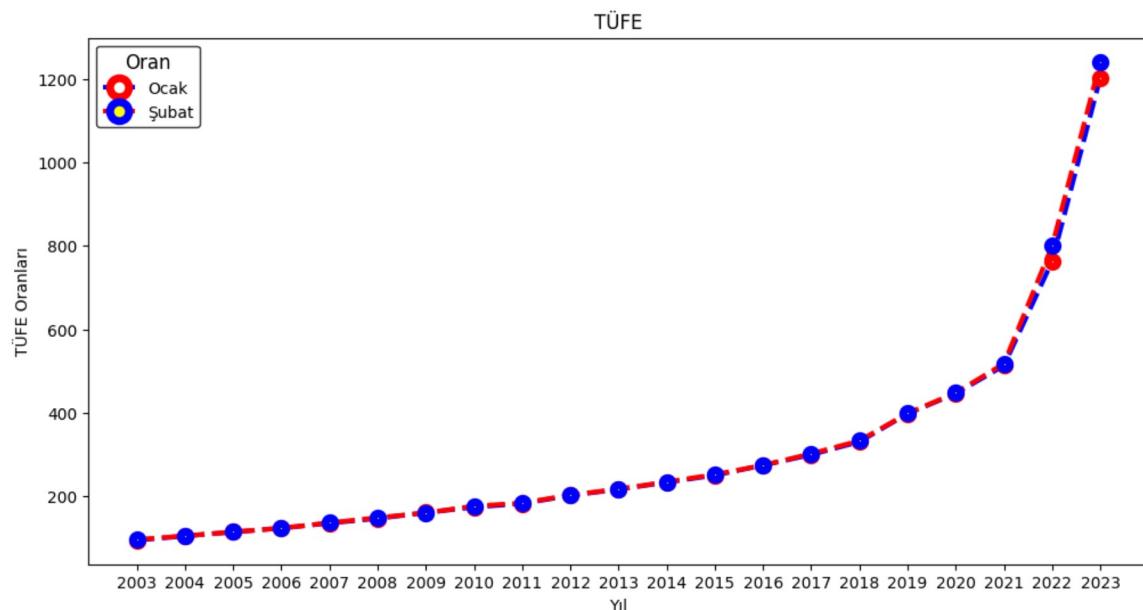


Farklı verileri, tek legend içinde gösterebiliriz.

```
In [ ]: plt.figure(figsize=(12,6))
plt.plot(data.index, data["Ocak"], color="blue", linewidth=3,
          linestyle="--", marker="o", markerfacecolor="white",
          markeredgewidth=5, markeredgecolor="red")

plt.plot(data.index, data["Şubat"], color="red", linewidth=3,
          linestyle="--", marker="o", markerfacecolor="yellow",
          markeredgewidth=5, markeredgecolor="blue")

plt.title("TÜFE")
plt.xlabel("Yıl")
plt.ylabel("TÜFE Oranları")
plt.legend(["Ocak", "Şubat"], loc=2, fontsize="medium",
           markerscale=2, edgecolor="black",
           title="Oran", title_fontsize=12)
plt.show()
```



Grafiğe Izgara Eklenmesi

bgrid, grafiğin arkasına grid(izgara) çizer.

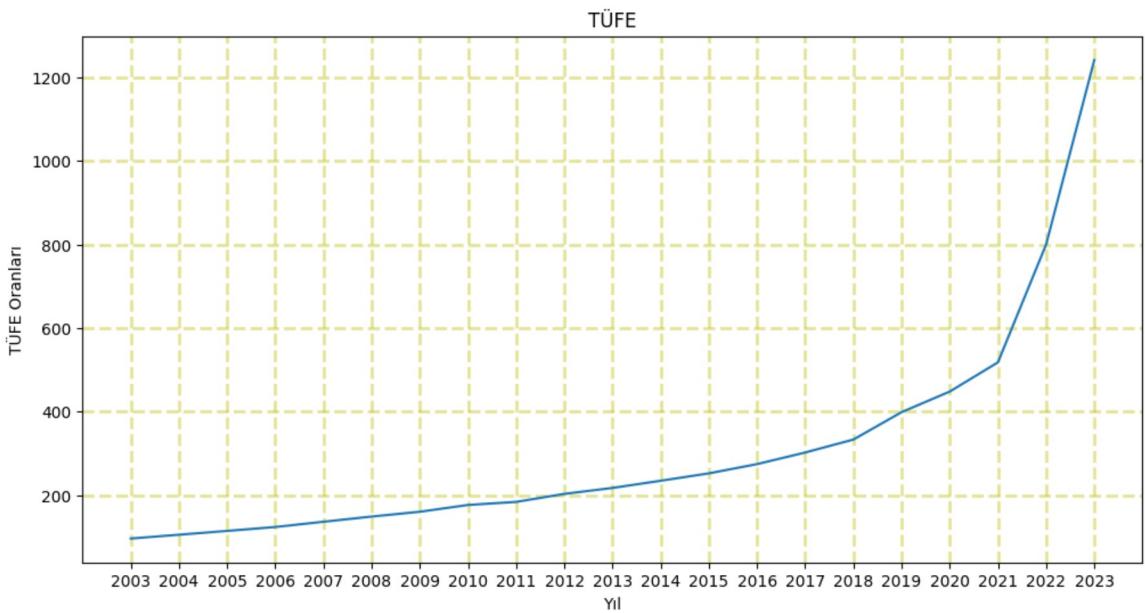
alpha, grid çizgilerinin saydamlığını ayarlar.0-1 arasında değer alır. 1'e yaklaşıkça saydamlık artar.0'a yaklaşıkça azalır.

linestyle, grid çizgilerinin biçimini belirler. - , -- , -. , : , gibi değerler alabilir.

linewidth, grid çubuklarının kalınlığını belirler. Integer değerler alır.

color, grid çizgilerinin rengini belirler.

```
In [ ]: plt.figure(figsize=(12,6))
plt.plot(data.index, data["Şubat"])
plt.title("TÜFE")
plt.xlabel("Yıl")
plt.ylabel("TÜFE Oranları")
plt.grid(alpha=0.4, linestyle="--", linewidth=2, color="y")
plt.show()
```



Grafiğe Üzerine Yazı Eklemek

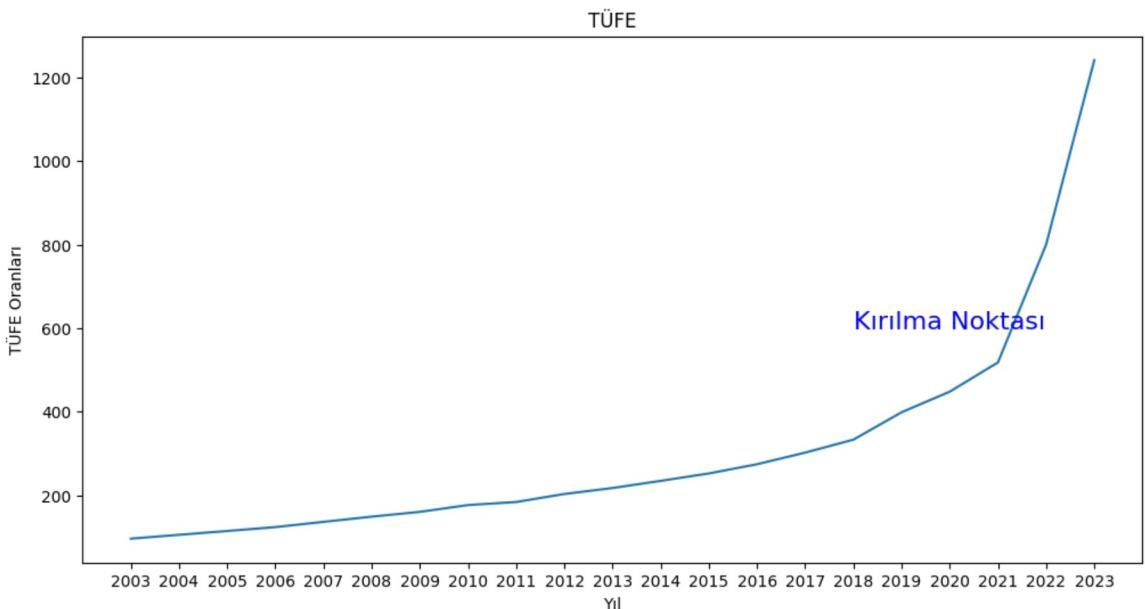
`plt.text()`, grafik üzerine yazı eklememizi sağlar.

`plt.text(x,y)`, yazının x ve y eksenine göre koordinatlarını belirtmemiz gereklidir.

`plt.text(x,y,s="")`, yazının belirtilmesidir.

`plt.text(x,y,s="",fontdict={"color":"r","fontsize":35})`, yazı ile ilgili düzeltmeler yapmamızı sağlar. Sözlük formatında kullanılır.

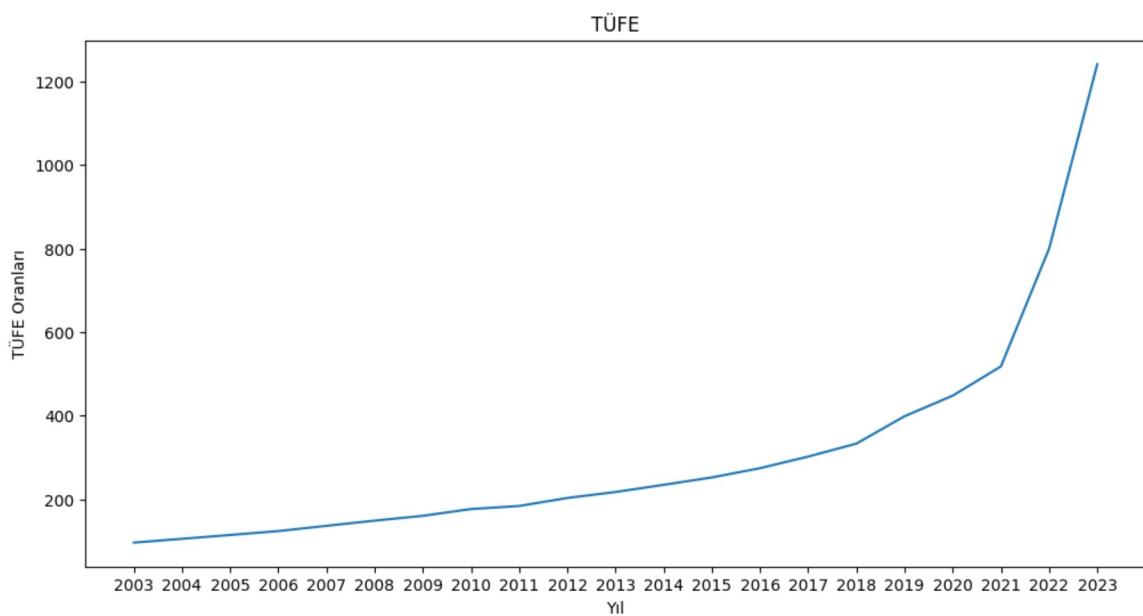
```
In [ ]: plt.figure(figsize=(12,6))
plt.plot(data.index, data["Şubat"])
plt.title("TÜFE")
plt.xlabel("Yıl")
plt.ylabel("TÜFE Oranları")
plt.text(x=15,y=600,s="Kırılma Noktası",fontdict={"color":"b","fontsize":16})
plt.show()
```



Grafiğin Kayıt Edilmesi

`plt.savefig()`, oluşturduğumuz grafiği kaydetmemizi sağlar.

```
In [ ]: plt.figure(figsize=(12,6))
plt.plot(data.index, data["Şubat"])
plt.title("TÜFE")
plt.xlabel("Yıl")
plt.ylabel("TÜFE Oranları")
plt.savefig("Şubat_Tüfe.png")
plt.show()
```



İletişim



Referans

- [Python İle Veri Görselleştirme | Matplotlib Kütüphanesi](#)

Berkay Yürür