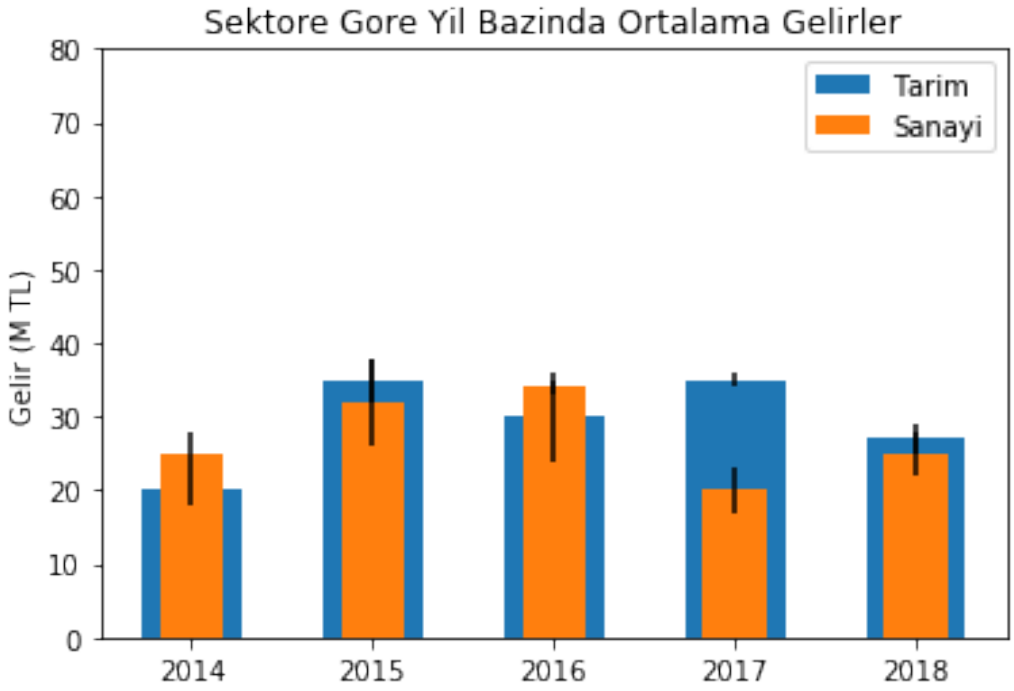
```
import matplotlib
import matplotlib.pyplot as plt
import numpy as np

N = 5
tarimOrt = (20, 35, 30, 35, 27)
sanayiOrt = (25, 32, 34, 20, 25)
tarimStd = (2, 3, 6, 1, 2)
sanayiStd = (3, 6, 1, 3, 3)

ind = np.arange(N)    # 0 dan 4 e kadar sayi dizisi (5 elemanlı)
print(ind)
genislik = 0.35       # çubuklarımızın uzunluğu

p1 = plt.bar(ind, tarimOrt, genislik+0.2, yerr=tarimStd)
p2 = plt.bar(ind, sanayiOrt, genislik,
yerr=sanayiStd)#bottom=tarimOrt,

plt.ylabel('Gelir (M TL)')
plt.title('Sektore Gore Yil Bazinda Ortalama Gelirler')
plt.xticks(ind, ('2014', '2015', '2016', '2017', '2018'))
plt.yticks(np.arange(0, 80+1, 10))
plt.legend((p1[0], p2[0]), ('Tarim', 'Sanayi'))

plt.show()
```

```
[0 1 2 3 4]
```



```
import numpy as np
import matplotlib.pyplot as plt
```

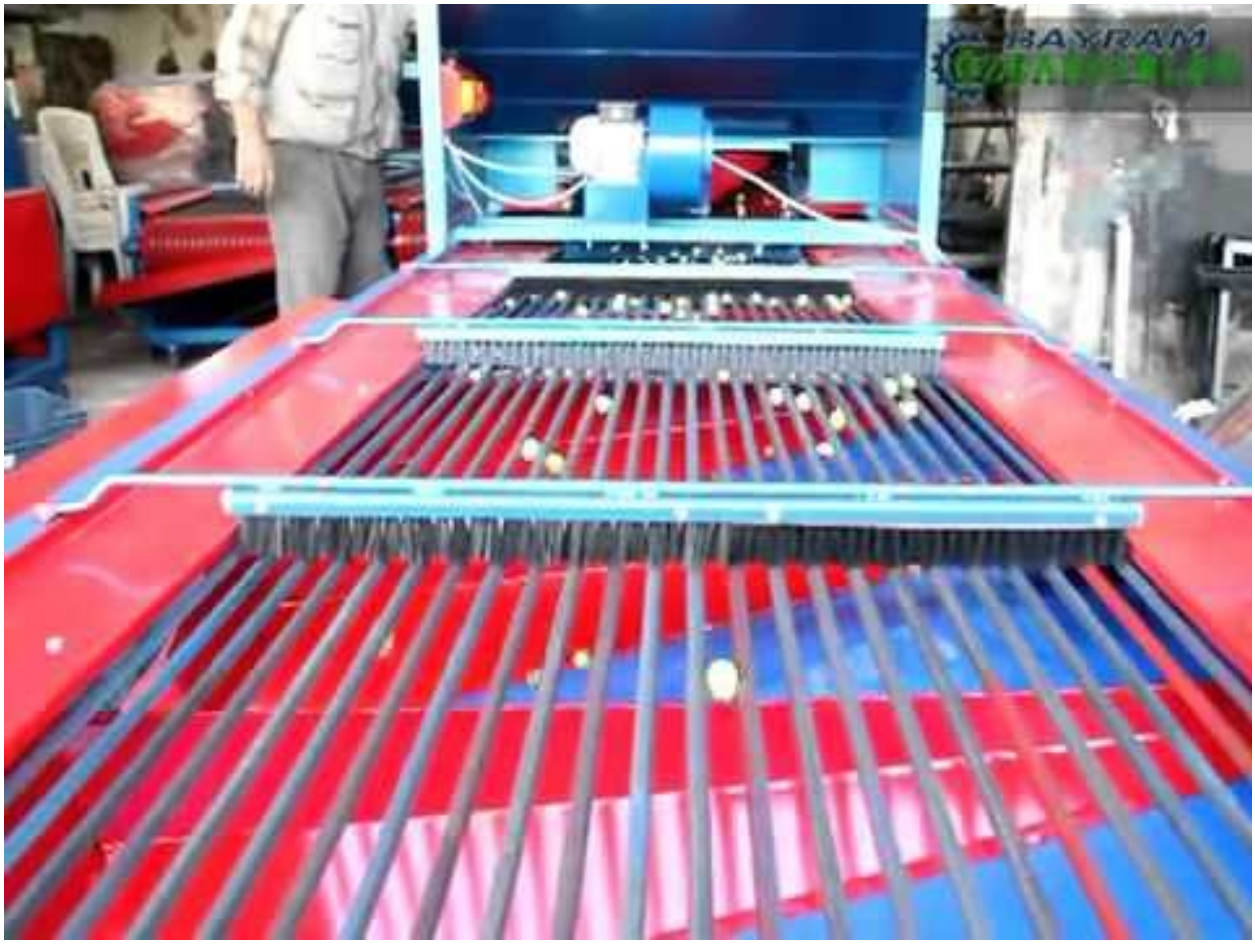
Veri Görselleme 2

Çubuk Garfikler

Histogramlar

Elimizdeki sayısal verinin dağılımını görsellemek için çubuk grafik (Histogram) uygun tekniklerden biridir.

Veriler teker teker değerlendirilip belirli 'sepet' lere yerleştirilirler. Sürecin sonunda hangi sepette kaç adet eleman olduğu sayılır.

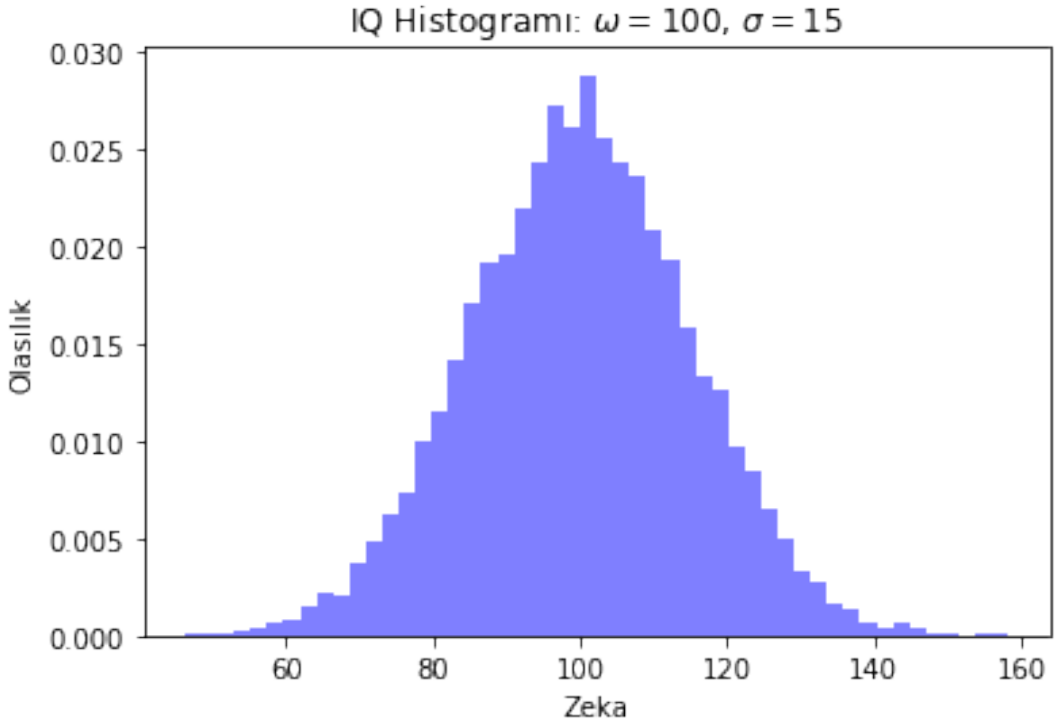


```
import numpy as np
import matplotlib.mlab as mlab
```

```
import matplotlib.pyplot as plt



# Verimizi üretelim
mu = 100 # verimizin ortalaması
sigma = 15 # standart sapması
x = mu + sigma * np.random.randn(10000)

num_bins = 50
# verimizin histogrami
n, bins, patches = plt.hist(x, num_bins, density=1, facecolor='blue',
alpha=0.5)

print(n)
print("---")
print(bins)
print("---")
print(patches)
print("---")
print(patches[1])

plt.xlabel('Zeka')
plt.ylabel('Olasılık')
plt.title(r'IQ Histogramı: $\omega=100$, $\sigma=15$')

# Sola yapismaya mani olmak icin eklenebilir
#plt.subplots_adjust(left=0.15, top = 0.9)
plt.show()
```

```
[4.46543103e-05 4.46543103e-05 1.78617241e-04 2.67925862e-04
 4.46543103e-04 6.25160344e-04 8.03777585e-04 1.47359224e-03
 2.23271551e-03 2.09875258e-03 3.66165344e-03 4.82266551e-03
 6.29625775e-03 7.27865258e-03 1.00472198e-02 1.15654664e-02
 1.41554164e-02 1.70132922e-02 1.92013534e-02 1.96032422e-02
 2.18806120e-02 2.42919448e-02 2.72391293e-02 2.61674258e-02
 2.88020301e-02 2.55422655e-02 2.42472905e-02 2.35774758e-02
 2.08089086e-02 1.93353164e-02 1.58969345e-02 1.33516388e-02
 1.25925155e-02 9.68998533e-03 8.48431896e-03 6.51952930e-03
 4.95662844e-03 3.25976465e-03 2.72391293e-03 1.60755517e-03
 1.33962931e-03 6.69814654e-04 4.01888793e-04 6.25160344e-04
 4.01888793e-04 4.46543103e-05 4.46543103e-05 0.00000000e+00
 8.93086206e-05 8.93086206e-05]
---
[ 46.24566981  48.48509528  50.72452076  52.96394623  55.2033717
  57.44279718  59.68222265  61.92164813  64.1610736   66.40049907
  68.63992455  70.87935002  73.11877549  75.35820097  77.59762644
  79.83705192  82.07647739  84.31590286  86.55532834  88.79475381
  91.03417929  93.27360476  95.51303023  97.75245571  99.99188118
 102.23130665 104.47073213 106.7101576  108.94958308 111.18900855
```

```
 113.42843402 115.6678595  117.90728497 120.14671045 122.38613592
 124.62556139 126.86498687 129.10441234 131.34383781 133.58326329
 135.82268876 138.06211424 140.30153971 142.54096518 144.78039066
 147.01981613 149.25924161 151.49866708 153.73809255 155.97751803
 158.2169435 ]
---
<a list of 50 Patch objects>
---
Rectangle(xy=(48.4851, 0), width=2.23943, height=4.46543e-05, angle=0)
```

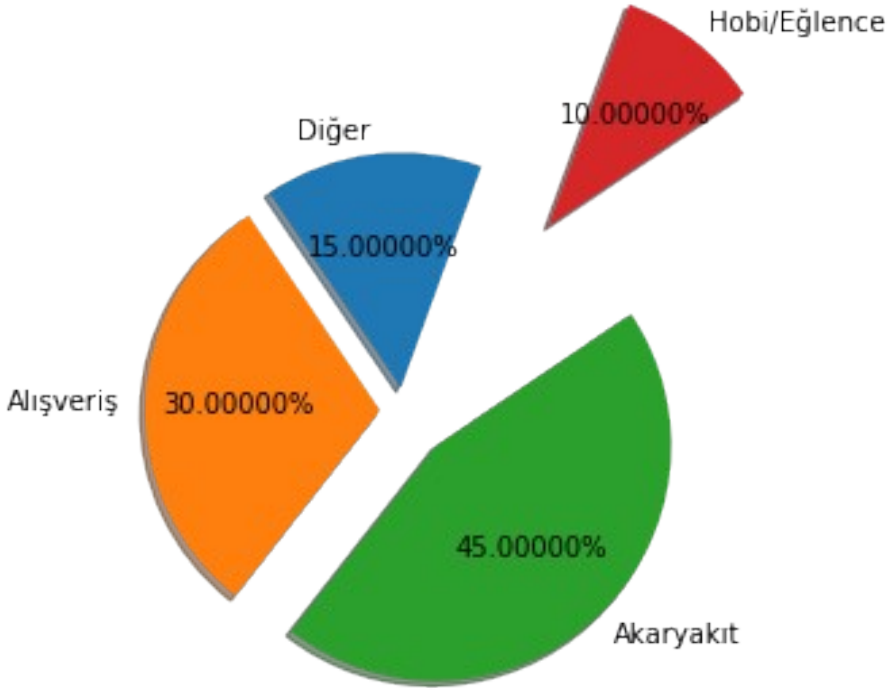


https://matplotlib.org/api/_as_gen/matplotlib.pyplot.subplots_adjust.html

Dilim Grafikler

```
import matplotlib.pyplot as plt

# Saat yönünün tersine çizilen dilim garfik:
labels = 'Diğer', 'Alışveriş', 'Akaryakıt', 'Hobi/Eğlence'
sizes = [15, 30, 45, 10]
explode = (0.1, 0.1, 0.2, 1)  # sadece istediğimiz dilimleri
ayırabiliriz

fig1, ax1 = plt.subplots()
ax1.pie(sizes, explode=explode, labels=labels, autopct='%1.5f%%',
        shadow=True, startangle=70)
#ax1.axis('equal')  # Dairenin yuvarlak çıkmasını sağlar.
```

```
plt.show()
```



http://detexify.kirelabs.org/classify.html