

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.datasets import load_iris
from sklearn.cluster import KMeans
```

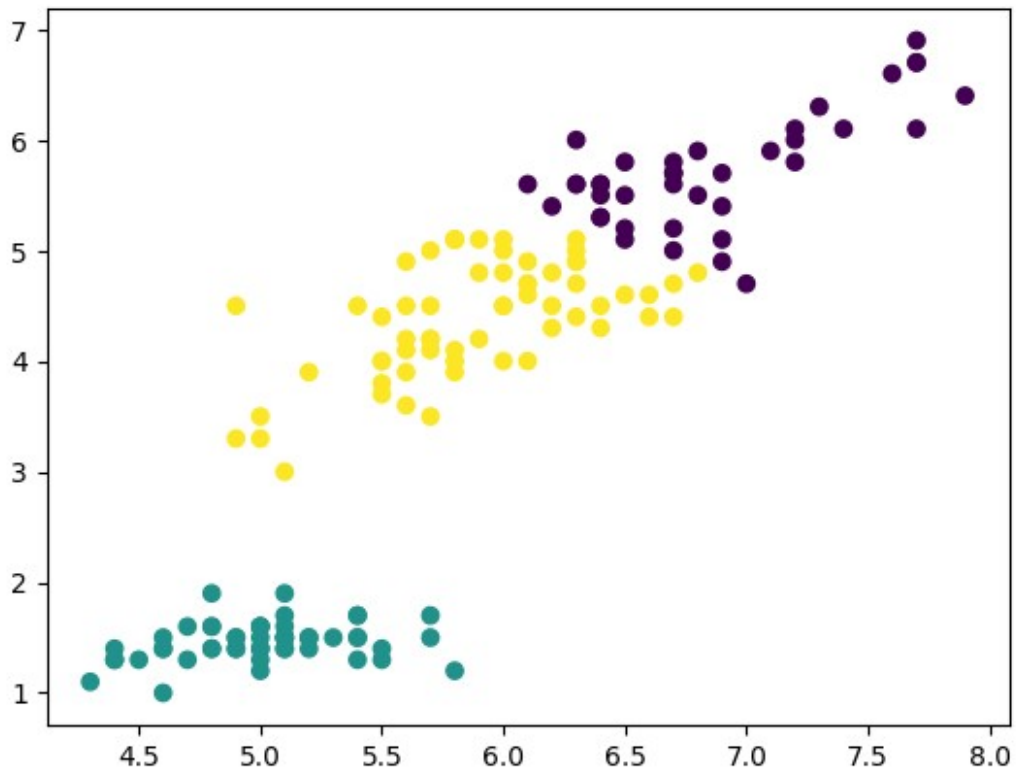
```
iris = load_iris()
print(iris.data)
```

```
[[5.1 3.5 1.4 0.2]
 [4.9 3.  1.4 0.2]
 [4.7 3.2 1.3 0.2]
 [4.6 3.1 1.5 0.2]
 [5.  3.6 1.4 0.2]
 [5.4 3.9 1.7 0.4]
 [4.6 3.4 1.4 0.3]
 [5.  3.4 1.5 0.2]
 [4.4 2.9 1.4 0.2]
 [4.9 3.1 1.5 0.1]
 [5.4 3.7 1.5 0.2]
 [4.8 3.4 1.6 0.2]
 [4.8 3.  1.4 0.1]
 [4.3 3.  1.1 0.1]
 [5.8 4.  1.2 0.2]
 [5.7 4.4 1.5 0.4]
 [5.4 3.9 1.3 0.4]
 [5.1 3.5 1.4 0.3]
 [5.7 3.8 1.7 0.3]
 [5.1 3.8 1.5 0.3]
 [5.4 3.4 1.7 0.2]
 [5.1 3.7 1.5 0.4]
 [4.6 3.6 1.  0.2]
 [5.1 3.3 1.7 0.5]
 [4.8 3.4 1.9 0.2]
 [5.  3.  1.6 0.2]
 [5.  3.4 1.6 0.4]
 [5.2 3.5 1.5 0.2]
 [5.2 3.4 1.4 0.2]
 [4.7 3.2 1.6 0.2]
 [4.8 3.1 1.6 0.2]
 [5.4 3.4 1.5 0.4]
 [5.2 4.1 1.5 0.1]
 [5.5 4.2 1.4 0.2]
 [4.9 3.1 1.5 0.2]
 [5.  3.2 1.2 0.2]
 [5.5 3.5 1.3 0.2]
 [4.9 3.6 1.4 0.1]
 [4.4 3.  1.3 0.2]
 [5.1 3.4 1.5 0.2]]
```

[5. 3.5 1.3 0.3]  
[4.5 2.3 1.3 0.3]  
[4.4 3.2 1.3 0.2]  
[5. 3.5 1.6 0.6]  
[5.1 3.8 1.9 0.4]  
[4.8 3. 1.4 0.3]  
[5.1 3.8 1.6 0.2]  
[4.6 3.2 1.4 0.2]  
[5.3 3.7 1.5 0.2]  
[5. 3.3 1.4 0.2]  
[7. 3.2 4.7 1.4]  
[6.4 3.2 4.5 1.5]  
[6.9 3.1 4.9 1.5]  
[5.5 2.3 4. 1.3]  
[6.5 2.8 4.6 1.5]  
[5.7 2.8 4.5 1.3]  
[6.3 3.3 4.7 1.6]  
[4.9 2.4 3.3 1. ]  
[6.6 2.9 4.6 1.3]  
[5.2 2.7 3.9 1.4]  
[5. 2. 3.5 1. ]  
[5.9 3. 4.2 1.5]  
[6. 2.2 4. 1. ]  
[6.1 2.9 4.7 1.4]  
[5.6 2.9 3.6 1.3]  
[6.7 3.1 4.4 1.4]  
[5.6 3. 4.5 1.5]  
[5.8 2.7 4.1 1. ]  
[6.2 2.2 4.5 1.5]  
[5.6 2.5 3.9 1.1]  
[5.9 3.2 4.8 1.8]  
[6.1 2.8 4. 1.3]  
[6.3 2.5 4.9 1.5]  
[6.1 2.8 4.7 1.2]  
[6.4 2.9 4.3 1.3]  
[6.6 3. 4.4 1.4]  
[6.8 2.8 4.8 1.4]  
[6.7 3. 5. 1.7]  
[6. 2.9 4.5 1.5]  
[5.7 2.6 3.5 1. ]  
[5.5 2.4 3.8 1.1]  
[5.5 2.4 3.7 1. ]  
[5.8 2.7 3.9 1.2]  
[6. 2.7 5.1 1.6]  
[5.4 3. 4.5 1.5]  
[6. 3.4 4.5 1.6]  
[6.7 3.1 4.7 1.5]  
[6.3 2.3 4.4 1.3]  
[5.6 3. 4.1 1.3]

[5.5 2.5 4. 1.3]  
[5.5 2.6 4.4 1.2]  
[6.1 3. 4.6 1.4]  
[5.8 2.6 4. 1.2]  
[5. 2.3 3.3 1. ]  
[5.6 2.7 4.2 1.3]  
[5.7 3. 4.2 1.2]  
[5.7 2.9 4.2 1.3]  
[6.2 2.9 4.3 1.3]  
[5.1 2.5 3. 1.1]  
[5.7 2.8 4.1 1.3]  
[6.3 3.3 6. 2.5]  
[5.8 2.7 5.1 1.9]  
[7.1 3. 5.9 2.1]  
[6.3 2.9 5.6 1.8]  
[6.5 3. 5.8 2.2]  
[7.6 3. 6.6 2.1]  
[4.9 2.5 4.5 1.7]  
[7.3 2.9 6.3 1.8]  
[6.7 2.5 5.8 1.8]  
[7.2 3.6 6.1 2.5]  
[6.5 3.2 5.1 2. ]  
[6.4 2.7 5.3 1.9]  
[6.8 3. 5.5 2.1]  
[5.7 2.5 5. 2. ]  
[5.8 2.8 5.1 2.4]  
[6.4 3.2 5.3 2.3]  
[6.5 3. 5.5 1.8]  
[7.7 3.8 6.7 2.2]  
[7.7 2.6 6.9 2.3]  
[6. 2.2 5. 1.5]  
[6.9 3.2 5.7 2.3]  
[5.6 2.8 4.9 2. ]  
[7.7 2.8 6.7 2. ]  
[6.3 2.7 4.9 1.8]  
[6.7 3.3 5.7 2.1]  
[7.2 3.2 6. 1.8]  
[6.2 2.8 4.8 1.8]  
[6.1 3. 4.9 1.8]  
[6.4 2.8 5.6 2.1]  
[7.2 3. 5.8 1.6]  
[7.4 2.8 6.1 1.9]  
[7.9 3.8 6.4 2. ]  
[6.4 2.8 5.6 2.2]  
[6.3 2.8 5.1 1.5]  
[6.1 2.6 5.6 1.4]  
[7.7 3. 6.1 2.3]  
[6.3 3.4 5.6 2.4]  
[6.4 3.1 5.5 1.8]

[illegible]



```
pointsDF = pd.read_csv("points.csv",index_col=0)

numpy_array = pointsDF.to_numpy()
print(numpy_array)

plt.figure()
sns.scatterplot(x=numpy_array[:,0],y=numpy_array[:,1])
plt.show()

xs = numpy_array[:,0]
ys = numpy_array[:,1]
plt.scatter(xs,ys)
plt.show()

[[ 0.06544649 -0.76866376]
 [-1.52901547 -0.42953079]
 [ 1.70993371  0.69885253]
 [ 1.16779145  1.01262638]
 [-1.80110088 -0.31861296]
 [-1.63567888 -0.02859535]
 [ 1.21990375  0.74643463]
 [-0.26175155 -0.62492939]
 [-1.61925804 -0.47983949]
 [-1.84329582 -0.16694431]
 [ 1.35999602  0.94995827]]
```

[ 0.42291856 -0.7349534 ]  
[ -1.68576139 0.10686728]  
[ 0.90629995 1.09105162]  
[ -1.56478322 -0.84675394]  
[ -0.0257849 -1.18672539]  
[ 0.83027324 1.14504612]  
[ 1.22450432 1.35066759]  
[ -0.15394596 -0.71704301]  
[ 0.86358809 1.06824613]  
[ -1.43386366 -0.2381297 ]  
[ 0.03844769 -0.74635022]  
[ -1.58567922 0.08499354]  
[ 0.6359888 -0.58477698]  
[ 0.24417242 -0.53172465]  
[ -2.19680359 0.49473677]  
[ 1.0323503 -0.55688 ]  
[ -0.28858067 -0.39972528]  
[ 0.20597008 -0.80171536]  
[ -1.2107308 -0.34924109]  
[ 1.33423684 0.7721489 ]  
[ 1.19480152 1.04788556]  
[ 0.9917477 0.89202008]  
[ -1.8356219 -0.04839732]  
[ 0.08415721 -0.71564326]  
[ -1.48970175 -0.19299604]  
[ 0.38782418 -0.82060119]  
[ -0.01448044 -0.9779841 ]  
[ -2.0521341 -0.02129125]  
[ 0.10331194 -0.82162781]  
[ -0.44189315 -0.65710974]  
[ 1.10390926 1.02481182]  
[ -1.59227759 -0.17374038]  
[ -1.47344152 -0.02202853]  
[ -1.35514704 0.22971067]  
[ 0.0412337 -1.23776622]  
[ 0.4761517 -1.13672124]  
[ 1.04335676 0.82345905]  
[ -0.07961882 -0.85677394]  
[ 0.87065059 1.08052841]  
[ 1.40267313 1.07525119]  
[ 0.80111157 1.28342825]  
[ -0.16527516 -1.23583804]  
[ -0.33779221 -0.59194323]  
[ 0.80610749 -0.73752159]  
[ -1.43590032 -0.56384446]  
[ 0.54868895 -0.95143829]  
[ 0.46803131 -0.74973907]  
[ -1.5137129 -0.83914323]  
[ 0.9138436 1.51126532]

[ -1.97233903 -0.41155375]  
[ 0.5213406 -0.88654894]  
[ 0.62759494 -1.18590477]  
[ 0.94163014 1.35399335]  
[ 0.56994768 1.07036606]  
[ -1.87663382 0.14745773]  
[ 0.90612186 0.91084011]  
[ -1.37481454 0.28428395]  
[ -1.80564029 -0.96710574]  
[ 0.34307757 -0.79999275]  
[ 0.70380566 1.00025804]  
[ -1.68489862 -0.30564595]  
[ 1.31473221 0.98614978]  
[ 0.26151216 -0.26069251]  
[ 0.9193121 0.82371485]  
[ -1.21795929 -0.20219674]  
[ -0.17722723 -1.02665245]  
[ 0.64824862 -0.66822881]  
[ 0.41206786 -0.28783784]  
[ 1.01568202 1.13481667]  
[ 0.67900254 -0.91489502]  
[ -1.05182747 -0.01062376]  
[ 0.61306599 1.78210384]  
[ -1.50219748 -0.52308922]  
[ -1.72717293 -0.46173916]  
[ -1.60995631 -0.1821007 ]  
[ -1.09111021 -0.0781398 ]  
[ -0.01046978 -0.80913034]  
[ 0.32782303 -0.80734754]  
[ 1.22038503 1.1959793 ]  
[ -1.33328681 -0.30001937]  
[ 0.87959517 1.11566491]  
[ -1.14829098 -0.30400762]  
[ -0.58019755 -1.19996018]  
[ -0.01161159 -0.78468854]  
[ 0.17359724 -0.63398145]  
[ 1.32738556 0.67759969]  
[ -1.93467327 0.30572472]  
[ -1.57761893 -0.27726365]  
[ 0.47639 1.21422648]  
[ -1.65237509 -0.6803981 ]  
[ -0.12609976 -1.04327457]  
[ -1.89607082 -0.70085502]  
[ 0.57466899 0.74878369]  
[ -0.16660312 -0.83110295]  
[ 0.8013355 1.22244435]  
[ 1.18455426 1.4346467 ]  
[ 1.08864428 0.64667112]  
[ -1.61158505 0.22805725]

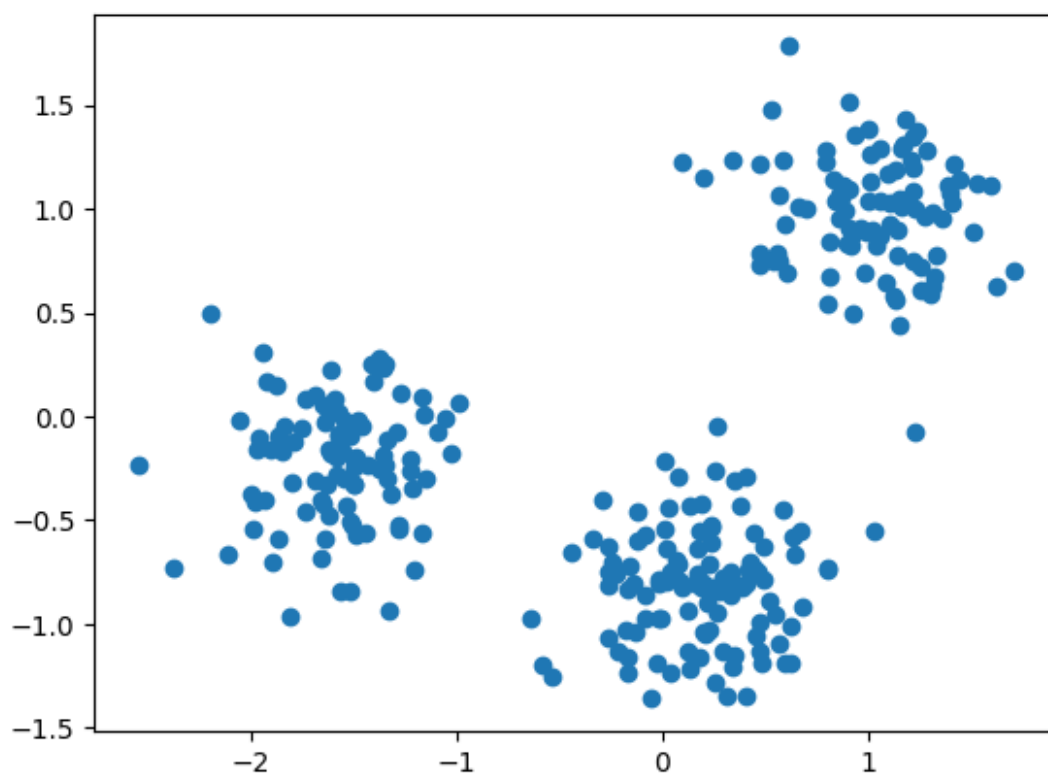
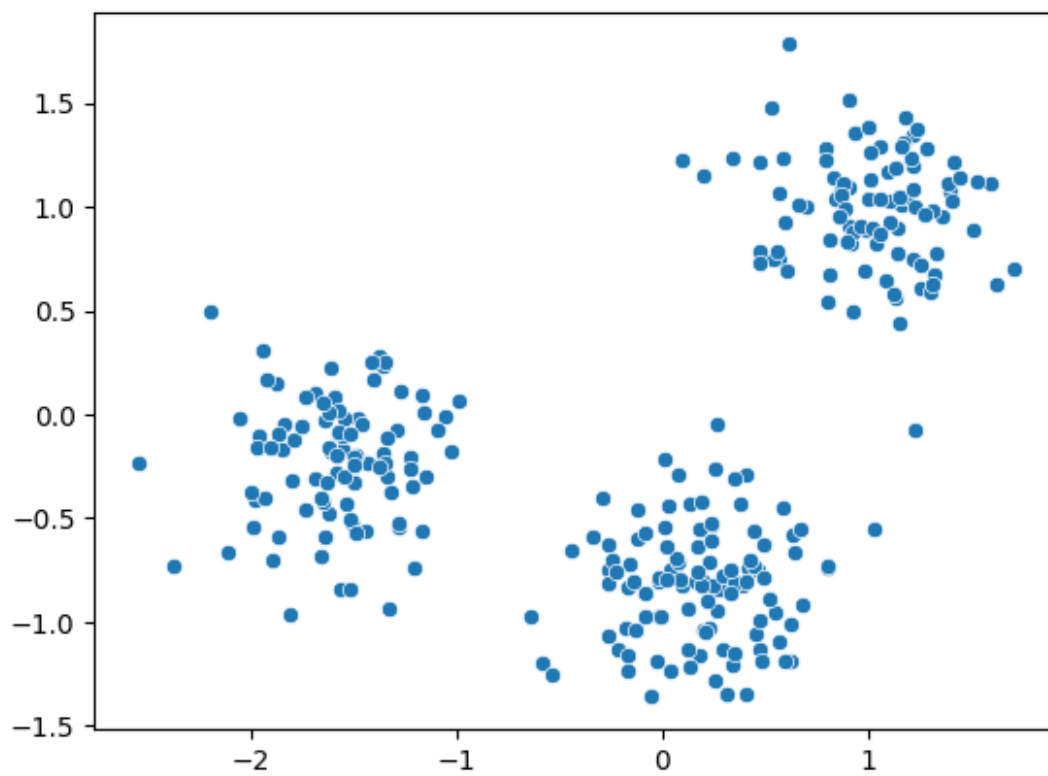
[ -1.57512205 -0.09612576]  
[ 0.0721357 -0.69640328]  
[ -1.40054298 0.16390598]  
[ 1.09607713 1.16804691]  
[ -2.54346204 -0.23089822]  
[ -1.34544875 0.25151126]  
[ -1.35478629 -0.19103317]  
[ 0.18368113 -1.15827725]  
[ -1.31368677 -0.376357 ]  
[ 0.09990129 1.22500491]  
[ 1.17225574 1.30835143]  
[ 0.0865397 -0.79714371]  
[ -0.21053923 -1.13421511]  
[ 0.26496024 -0.94760742]  
[ -0.2557591 -1.06266022]  
[ -0.26039757 -0.74774225]  
[ -1.91787359 0.16434571]  
[ 0.93021139 0.49436331]  
[ 0.44770467 -0.72877918]  
[ -1.63802869 -0.58925528]  
[ -1.95712763 -0.10125137]  
[ 0.9270337 0.88251423]  
[ 1.25660093 0.60828073]  
[ -1.72818632 0.08416887]  
[ 0.3499788 -0.30490298]  
[ -1.51696082 -0.50913109]  
[ 0.18763605 -0.55424924]  
[ 0.89609809 0.83551508]  
[ -1.54968857 -0.17114782]  
[ 1.2157457 1.23317728]  
[ 0.20307745 -1.03784906]  
[ 0.84589086 1.03615273]  
[ 0.53237919 1.47362884]  
[ -0.05319044 -1.36150553]  
[ 1.38819743 1.11729915]  
[ 1.00696304 1.0367721 ]  
[ 0.56681869 -1.09637176]  
[ 0.86888296 1.05248874]  
[ -1.16286609 -0.55875245]  
[ 0.27717768 -0.83844015]  
[ 0.16563267 -0.80306607]  
[ 0.38263303 -0.42683241]  
[ 1.14519807 0.89659026]  
[ 0.81455857 0.67533667]  
[ -1.8603152 -0.09537561]  
[ 0.965641 0.90295579]  
[ -1.49897451 -0.33254044]  
[ -0.1335489 -0.80727582]  
[ 0.12541527 -1.13354906]



[ 1.06062436 1.28816358]  
[ -1.49154578 -0.2024641 ]  
[ 1.16189032 1.28819877]  
[ 0.54282033 0.75203524]  
[ 0.89221065 0.99211624]  
[ -1.49932011 -0.32430667]  
[ 0.3166647 -1.34482915]  
[ 0.13972469 -1.22097448]  
[ -1.5499724 -0.10782584]  
[ 1.23846858 1.37668804]  
[ 1.25558954 0.72026098]  
[ 0.25558689 -1.28529763]  
[ 0.45168933 -0.55952093]  
[ 1.06202057 1.03404604]  
[ 0.67451908 -0.54970299]  
[ 0.22759676 -1.02729468]  
[ -1.45835281 -0.04951074]  
[ 0.23273501 -0.70849262]  
[ 1.59679589 1.11395076]  
[ 0.80476105 0.544627 ]  
[ 1.15492521 1.04352191]  
[ 0.59632776 -1.19142897]  
[ 0.02839068 -0.43829366]  
[ 1.13451584 0.5632633 ]  
[ 0.21576204 -1.04445753]  
[ 1.41048987 1.02830719]  
[ 1.12289302 0.58029441]  
[ 0.25200688 -0.82588436]  
[ -1.28566081 -0.07390909]  
[ 1.52849815 1.11822469]  
[ -0.23907858 -0.70541972]  
[ -0.25792784 -0.81825035]  
[ 0.59367818 -0.45239915]  
[ 0.07931909 -0.29233213]  
[ -1.27256815 0.11630577]  
[ 0.66930129 1.00731481]  
[ 0.34791546 -1.20822877]  
[ -2.11283993 -0.66897935]  
[ -1.6293824 -0.32718222]  
[ -1.53819139 -0.01501972]  
[ -0.11988545 -0.6036339 ]  
[ -1.54418956 -0.30389844]  
[ 0.30026614 -0.77723173]  
[ 0.00935449 -0.53888192]  
[ -1.33424393 -0.11560431]  
[ 0.47504489 0.78421384]  
[ 0.59313264 1.232239 ]  
[ 0.41370369 -1.35205857]  
[ 0.55840948 0.78831053]

[ 0.49855018 -0.789949 ]  
[ 0.35675809 -0.81038693]  
[ -1.86197825 -0.59071305]  
[ -1.61977671 -0.16076687]  
[ 0.80779295 -0.73311294]  
[ 1.62745775 0.62787163]  
[ -1.56993593 -0.08467567]  
[ 1.02558561 0.89383302]  
[ 0.24293461 -0.6088253 ]  
[ 1.23130242 1.00262186]  
[ -1.9651013 -0.15886289]  
[ 0.42795032 -0.70384432]  
[ -1.58306818 -0.19431923]  
[ -1.57195922 0.01413469]  
[ -0.98145373 0.06132285]  
[ -1.48637844 -0.5746531 ]  
[ 0.98745828 0.69188053]  
[ 1.28619721 1.28128821]  
[ 0.85850596 0.95541481]  
[ 0.19028286 -0.82112942]  
[ 0.26561046 -0.04255239]  
[ -1.61897897 0.00862372]  
[ 0.24070183 -0.52664209]  
[ 1.15220993 0.43916694]  
[ -1.21967812 -0.2580313 ]  
[ 0.33412533 -0.86117761]  
[ 0.17131003 -0.75638965]  
[ -1.19828397 -0.73744665]  
[ -0.12245932 -0.45648879]  
[ 1.51200698 0.88825741]  
[ 1.10338866 0.92347479]  
[ 1.30972095 0.59066989]  
[ 0.19964876 1.14855889]  
[ 0.81460515 0.84538972]  
[ -1.6422739 -0.42296206]  
[ 0.01224351 -0.21247816]  
[ 0.33709102 -0.74618065]  
[ 0.47301054 0.72712075]  
[ 0.34706626 1.23033757]  
[ -0.00393279 -0.97209694]  
[ -1.64303119 0.05276337]  
[ 1.44649625 1.14217033]  
[ -1.93030087 -0.40026146]  
[ -2.37296135 -0.72633645]  
[ 0.45860122 -1.06048953]  
[ 0.4896361 -1.18928313]  
[ -1.02335902 -0.17520578]  
[ -1.32761107 -0.93963549]  
[ -1.50987909 -0.09473658]

```
[ 0.02723057 -0.79870549]
[ 1.0169412  1.26461701]
[ 0.47733527 -0.9898471 ]
[-1.27784224 -0.547416  ]
[ 0.49898802 -0.6237259 ]
[ 1.06004731  0.86870008]
[ 1.00207501  1.38293512]
[ 1.31161394  0.62833956]
[ 1.13428443  1.18346542]
[ 1.27671346  0.96632878]
[-0.63342885 -0.97768251]
[ 0.12698779 -0.93142317]
[-1.34510812 -0.23754226]
[-0.53162278 -1.25153594]
[ 0.21959934 -0.90269938]
[-1.78997479 -0.12115748]
[ 1.23197473 -0.07453764]
[ 1.4163536   1.21551752]
[-1.90280976 -0.1638976  ]
[-0.22440081 -0.75454248]
[ 0.59559412  0.92414553]
[ 1.21930773  1.08175284]
[-1.99427535 -0.37587799]
[-1.27818474 -0.52454551]
[ 0.62352689 -1.01430108]
[ 0.14024251 -0.428266  ]
[-0.16145713 -1.16359731]
[-1.74795865 -0.06033101]
[-1.16659791  0.0902393  ]
[ 0.41110408 -0.8084249  ]
[ 1.14757168  0.77804528]
[-1.65590748 -0.40105446]
[-1.15306865  0.00858699]
[ 0.60892121  0.68974833]
[-0.08434138 -0.97615256]
[ 0.19170053 -0.42331438]
[ 0.29663162 -1.13357399]
[-1.36893628 -0.25052124]
[-0.08037807 -0.56784155]
[ 0.35695011 -1.15064408]
[ 0.02482179 -0.63594828]
[-1.49075558 -0.2482507  ]
[-1.408588    0.25635431]
[-1.98274626 -0.54584475]]
```



```

model = KMeans(n_clusters=3,random_state=42)
model.fit(numpy_array.data)

labels = model.predict(numpy_array.data)
print(labels)

[2 1 0 0 1 1 0 2 1 1 0 2 1 0 1 2 0 0 2 0 1 2 1 2 2 1 2 2 2 1 0 0 0 1 2
 1 2
 2 1 2 2 0 1 1 1 2 2 0 2 0 0 0 2 2 2 1 2 2 1 0 1 2 2 0 0 1 0 1 1 2 0 1
 0 2
 0 1 2 2 2 0 2 1 0 1 1 1 1 2 2 0 1 0 1 2 2 2 0 1 1 0 1 2 1 0 2 0 0 0 1
 1 2
 1 0 1 1 1 2 1 0 0 2 2 2 2 2 1 0 2 1 1 0 0 1 2 1 2 0 1 0 2 0 0 2 0 0 2
 0 1
 2 2 2 0 0 1 0 1 2 2 0 1 0 0 0 1 2 2 1 0 0 2 2 0 2 2 1 2 0 0 0 2 2 0 2
 0 0
 2 1 0 2 2 2 2 1 0 2 1 1 1 2 1 2 2 1 0 0 2 0 2 2 1 1 2 0 1 0 2 0 1 2 1
 1 1
 1 0 0 0 2 2 1 2 0 1 2 2 1 2 0 0 0 0 0 1 2 2 0 0 2 1 0 1 1 2 2 1 1 1 2
 0 2
 1 2 0 0 0 0 0 2 2 1 2 2 1 0 0 1 2 0 0 1 1 2 2 2 1 1 2 0 1 1 0 2 2 2 1
 2 2
 2 1 1 1]

new_pointsDF = pd.read_csv("new_points.csv",index_col=0)
new_pointsDF_array = new_pointsDF.to_numpy()

model = KMeans(n_clusters=3,random_state=42)
model.fit(new_pointsDF_array)

labels = model.predict(new_pointsDF_array)
print(labels)

plt.figure()
xs = new_pointsDF_array[:,0]
ys = new_pointsDF_array[:,1]
plt.scatter(xs,ys,c = labels,alpha=0.5)

plt.show()

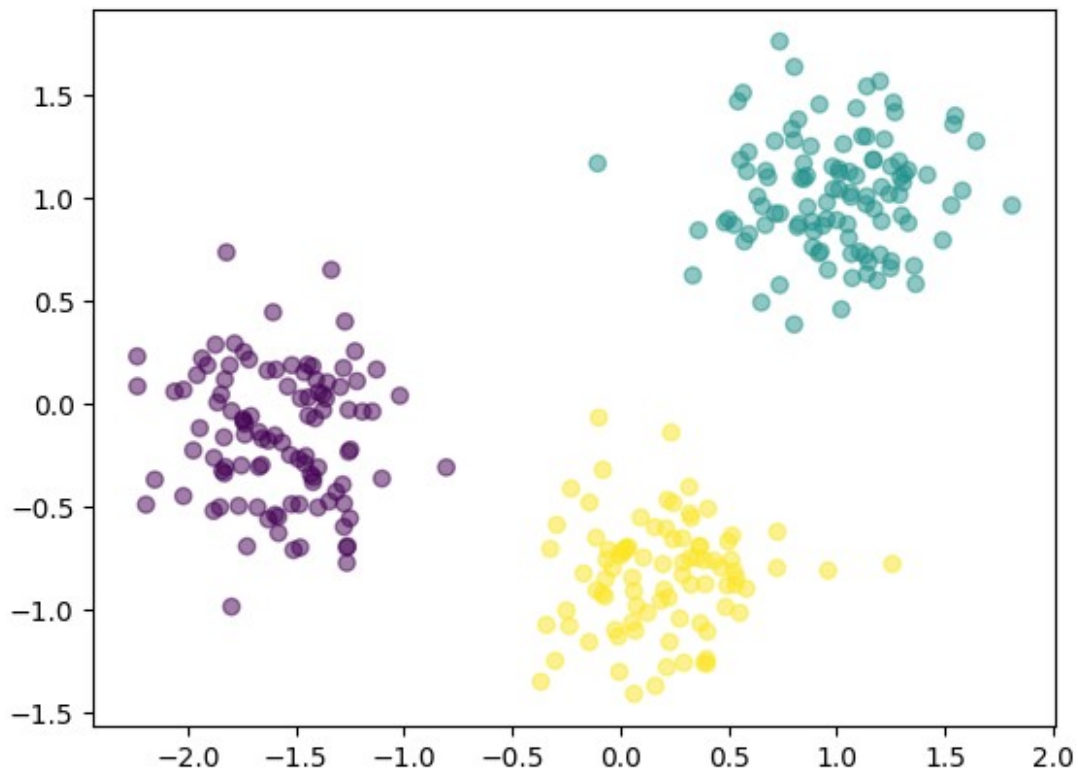
[2 1 0 2 1 2 1 1 1 0 2 1 1 0 0 1 0 0 1 1 0 1 2 1 2 0 1 0 0 2 2 1 1 1 0
 2 1
 1 2 1 0 2 2 0 2 1 0 0 1 1 1 1 0 0 2 2 0 0 0 2 2 1 1 1 2 1 0 1 2 0 2 2
 2 1
 2 0 0 2 1 0 2 0 2 1 0 1 0 2 1 1 1 2 1 1 2 0 0 0 0 2 1 2 0 0 2 2 1 2 0
 0 2
 0 0 0 1 1 1 1 0 0 1 2 1 0 1 2 0 1 0 0 1 0 1 0 2 1 2 2 1 0 2 1 2 2 0 1
 1 2
 0 2 0 1 2 0 0 2 0 1 1 0 1 0 0 1 1 2 1 1 0 2 0 2 2 1 2 1 1 2 2 0 2 2 2
 0 1

```

```

1 2 0 2 0 0 1 1 1 2 1 1 1 0 0 2 1 2 2 2 0 1 1 1 1 1 1 0 0 1 0 0 0 0 1
0 0
1 1 2 0 2 2 0 2 0 2 0 1 1 0 1 1 1 0 2 2 0 1 1 0 1 0 0 1 0 0 2 0 2 2 2
1 0
0 0 2 1 2 0 2 0 0 1 2 2 2 0 1 1 1 2 1 0 0 1 2 2 0 2 2 0 2 1 2 0 0 0 0
1 0
0 1 1 2]

```



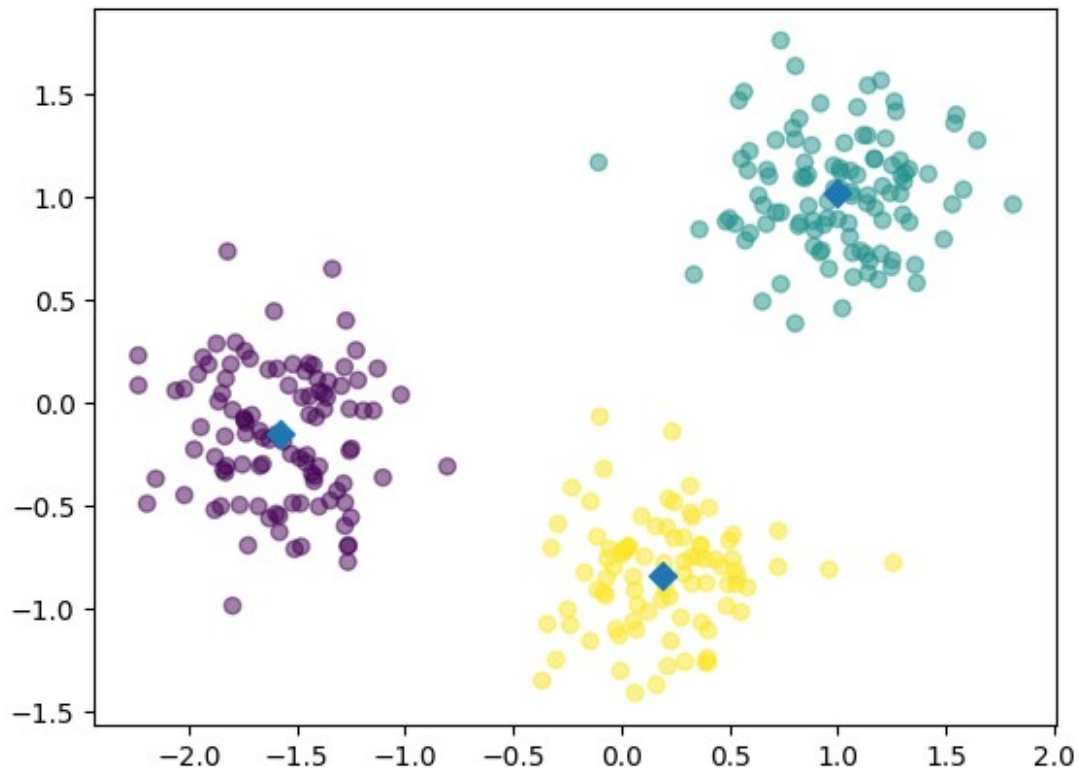
```

#plt.figure()
xs = new_pointsDF_array[:,0]
ys = new_pointsDF_array[:,1]
plt.scatter(xs,ys,c = labels,alpha=0.5)

centroids = model.cluster_centers_
centroids_x = centroids[:,0]
centroids_y = centroids[:,1]

plt.scatter(centroids_x,centroids_y,marker="D",s=50)
plt.show()

```

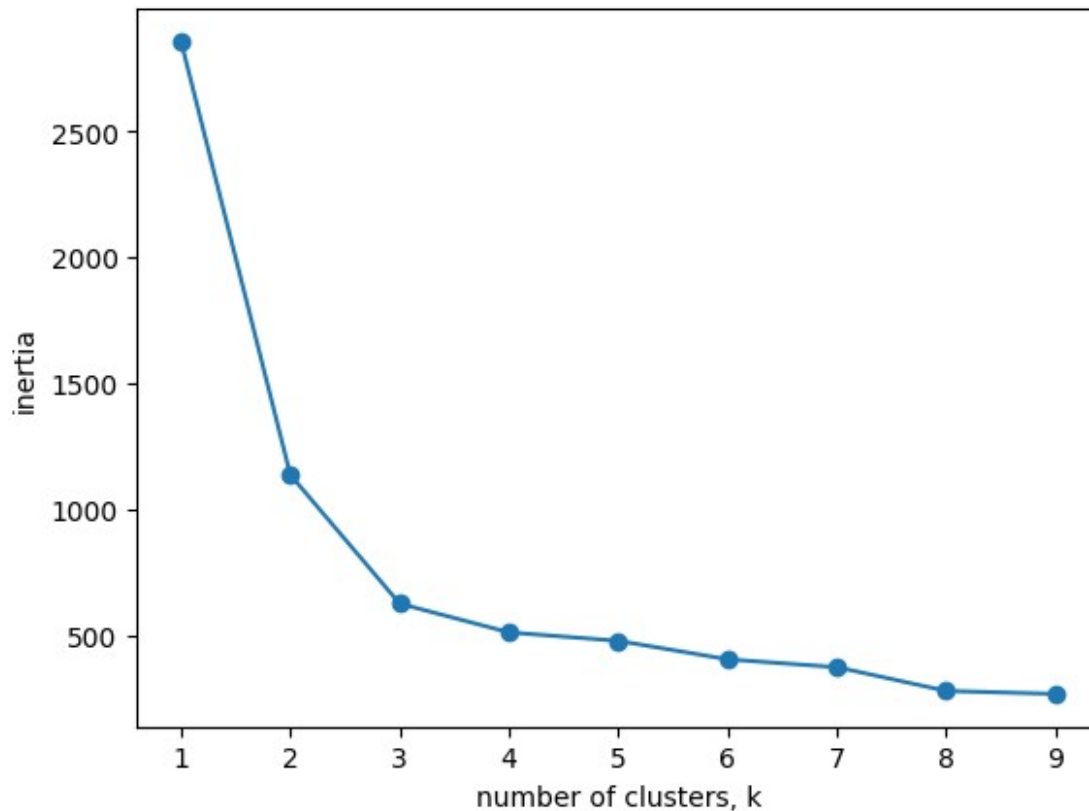


```
seedsDF = pd.read_csv('seeds.csv')
seeds = seedsDF.to_numpy()

ks = range(1,10)
inertias = []

for k in ks:
    model = KMeans(n_clusters=k,random_state=42)
    model.fit(seeds)
    inertias.append(model.inertia_)

plt.plot(ks,inertias,"-o")
plt.xlabel('number of clusters, k')
plt.ylabel('inertia')
plt.xticks(ks)
plt.show()
```



```
columns = ['area A', 'perimeter P', 'compactness C', 'length of
kernel', 'width of kernel', 'asymmetry coefficient', 'length of kernel
groove']
seedsDF = pd.read_csv('seeds.csv', names=columns)
seeds = seedsDF.to_numpy()
print(seeds)

[[14.84    0.871    5.763    ...    2.221    5.22    1.    ]
 [14.57    0.8811   5.554    ...    1.018    4.956    1.    ]
 [14.09    0.905     5.291    ...    2.699    4.825    1.    ]
 ...
 [13.66    0.8883    5.236    ...    8.315    5.056    3.    ]
 [13.21    0.8521    5.175    ...    3.598    5.044    3.    ]
 [13.34    0.8684    5.243    ...    5.637    5.063    3.    ]]

model = KMeans(n_clusters=3, random_state=42)
model.fit(seeds)

labels = model.predict(seeds)
print(labels)

x = seeds[:,0]
y = seeds[:,1]
plt.scatter(x,y,c=labels)
```





```

pipeline = make_pipeline(scaler,model)

species = ['Bream', 'Bream', 'Bream', 'Bream', 'Bream', 'Bream',
'Bream', 'Bream', 'Bream', 'Bream', 'Bream', 'Bream', 'Bream',
'Bream', 'Bream', 'Bream', 'Bream', 'Bream', 'Bream', 'Bream',
'Bream', 'Bream', 'Bream', 'Bream', 'Bream', 'Bream', 'Bream',
'Roach', 'Roach', 'Roach', 'Roach', 'Roach', 'Roach', 'Roach',
'Roach', 'Roach', 'Roach', 'Roach', 'Roach', 'Roach', 'Roach',
'Roach', 'Roach', 'Roach', 'Roach', 'Roach', 'Roach', 'Smelt',
'Smelt', 'Smelt', 'Smelt', 'Smelt', 'Smelt', 'Smelt', 'Smelt',
'Smelt', 'Smelt', 'Smelt', 'Smelt', 'Smelt', 'Smelt', 'Pike', 'Pike',
'Pike', 'Pike', 'Pike', 'Pike', 'Pike', 'Pike', 'Pike', 'Pike',
'Pike', 'Pike', 'Pike', 'Pike', 'Pike', 'Pike', 'Pike']
fish = pd.read_csv('fish.csv',header=None)
fish = fish.drop(0,axis=1)
labels = model.fit_predict(fish.to_numpy())

df = pd.DataFrame({"labels":labels,"species":species})
pd.crosstab(df["labels"],df["species"])

species  Bream  Pike  Roach  Smelt
labels
0          13    10     3     0
1           0     4     0     0
2           1     1    17    14
3          20     2     0     0

fish_arr = fish.to_numpy()
x = fish_arr[:,0]
y = fish_arr[:,1]

plt.scatter(x,y, c=labels)
plt.show()

```

