

# K-Means:

```
from sklearn.datasets import load_iris

from sklearn.cluster import KMeans

from sklearn.mixture import GaussianMixture

from sklearn.metrics import completeness_score

import matplotlib.pyplot as plt

import numpy as np


data=load_iris()

x=data.data

y=data.target


wcss=[ ]

for i in range(2,11):

    model=KMeans(n_clusters=i)

    model.fit(x)

    wcss.append(model.inertia_)

plt.figure()

plt.plot(range(2,11),wcss)


model=KMeans(n_clusters = 3)

model.fit(x)

print("the completeness score of KMeans is:", completeness_score(y,model.labels_))
```

```
gmm=GaussianMixture(n_components = 3,random_state=1)

gmm.fit(x)

y_pred=gmm.predict(x)

print("the completeness score of Gaussian Mixture is :", completeness_score(y,y_pred))
```

```
plt.figure(figsize=(21,7))

colorMap = np.array(["lime","red","black"])
```

```
plt.subplot(1,3,1)

plt.scatter(x[:,2],x[:,3],c=colorMap[y])

plt.title("real Classification")

plt.xlabel("Petal Length")

plt.ylabel("petal Width")

plt.subplot(1,3,2)

plt.scatter(x[:,2],x[:,3],c=colorMap[model.labels_])

plt.title("KMeans classification")

plt.xlabel("Petal Length")

plt.ylabel("petal Width")
```

```
plt.subplot(1,3,3)

plt.scatter(x[:,2],x[:,3],c=colorMap[gmm.predict(x)], s=40)

plt.title("Gaussian Mixture classification")

plt.xlabel("Petal Length")

plt.ylabel("petal Width")
```

```
Out[1]:
```

```
Text(0, 0.5, 'petal Width')
```

