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

from sklearn.datasets import make_blobs
from sklearn.cluster import DBSCAN

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

n_samples = 1500
random_state = 170
X, y = make_blobs(n_samples=n_samples, random_state=random_state)

# Anisotropicly distributed data
transformation = [[ 0.60834549, -0.63667341], [-0.40887718, 0.85253229]]
X_aniso = np.dot(X, transformation)

X_varied, y_varied = make_blobs(n_samples=n_samples,
                                cluster_std=[1.0, 2.5, 0.5],
                                random_state=random_state)

# Unevenly sized blobs
X_filtered = np.vstack((X[y == 0][:500], X[y == 1][:100], X[y == 2][:10]))

# Incorrect number of clusters
ya_pred = DBSCAN(eps=0.5, min_samples=8).fit_predict(X)

plt.scatter(X[:, 0], X[:, 1], c=ya_pred, s=100)
plt.title("Example - eps=0.5, min_samples=8")
plt.show()

yd1_pred = DBSCAN(eps=0.1, min_samples=8).fit_predict(X_aniso)
yd2_pred = DBSCAN(eps=0.2, min_samples=8).fit_predict(X_aniso)
yd3_pred = DBSCAN(eps=0.3, min_samples=8).fit_predict(X_aniso)
yd4_pred = DBSCAN(eps=0.4, min_samples=8).fit_predict(X_aniso)

fig = plt.figure(figsize=(12, 12))
ax2 = fig.add_subplot(2,2,1)
ax2.scatter(X_aniso[:,0],X_aniso[:,1], c=yd1_pred, s=50)
plt.title("Anisotropicly Distributed Blobs - eps=0.1")
ax1 = fig.add_subplot(2,2,2)
ax1.scatter(X_aniso[:,0],X_aniso[:,1], c=yd2_pred, s=50)
plt.title("Anisotropicly Distributed Blobs - eps=0.2")
ax3 = fig.add_subplot(2,2,3)
ax3.scatter(X_aniso[:,0],X_aniso[:,1], c=yd3_pred, s=50)
plt.title("Anisotropicly Distributed Blobs - eps=0.3")
ax4 = fig.add_subplot(2,2,4)
ax4.scatter(X_aniso[:,0],X_aniso[:,1], c=yd4_pred, s=50)
plt.title("Anisotropicly Distributed Blobs - eps=0.4")
plt.show()

fig = plt.figure(figsize=(12, 12))
for i in range(1,5):
    yd1_pred = DBSCAN(eps=0.3, min_samples=i+2).fit_predict(X_aniso)
    ax2 = fig.add_subplot(2,2,i)
    ax2.scatter(X_aniso[:,0],X_aniso[:,1], c=yd1_pred, s=50)
    plt.title("Anisotropicly Distributed Blobs - min_samples="+ str(i+2))
plt.show()


# Different variance
fig = plt.figure(figsize=(12, 12))
for i in range(1,5):
    v = 0.4+i*.1
    yd1_pred = DBSCAN(eps=v, min_samples=6).fit_predict(X_varied)
    ax2 = fig.add_subplot(2,2,i)
    ax2.scatter(X_varied[:,0],X_varied[:,1], c=yd1_pred, s=50)
    plt.title("Unequal Variance - eps="+ str(v))
```

```
plt.show()

# Unevenly Sized Blobs
fig = plt.figure(figsize=(12, 12))
for i in range(1,5):
    v = 1+i*.2
    yd1_pred = DBSCAN(eps=v, min_samples=2).fit_predict(X_filtered)
    ax2 = fig.add_subplot(2,2,i)
    ax2.scatter(X_filtered[:,0],X_filtered[:,1], c=yd1_pred, s=50)
    plt.title("Unevenly Sized Blobs - eps ="+ str(v))
plt.show()
```