

```
# Citation 1 : From Canvas Project Example (Jupyter Notebook, knn.synthetic_data.ipynb)
# Citation 2 : From Canvas Project Example (Jupyter Notebook, kmeans.synthetic_data.ipynb)
# Citation 3 : https://www.geeksforgeeks.org/3d-scatter-plotting-in-python-using-matplotlib/
```

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from scipy.spatial import distance
import collections

def euclidean_distance(X1, Y1, Z1, X2, Y2, Z2):

    distance = ( ((X2-X1)**2) + ((Y2-Y1)**2) + ((Z2-Z1)**2) )
    distance = sqrt(distance)

    return distance

def knn(newObservation, referenceData, k=3):
    knn(X_test.iloc[0], referenceData, k=3)

np.random.seed(42)

#data points 2D
n = 100

#draw from a Gaussian distribution

X0 = np.random.normal(loc=-2.0, scale=2.0, size=int(n/2))
Y0 = np.random.normal(loc=0.0, scale=1.0, size=int(n/2))

X1 = np.random.normal(loc=2.0, scale=2.0, size=int(n/2))
Y1 = np.random.normal(loc=0.0, scale=1.0, size=int(n/2))

X = np.concatenate((X0, X1), axis=0)
Y = np.concatenate((Y0, Y1), axis=0)

dt = pd.DataFrame({'X':X, 'Y':Y}, columns=['X', 'Y'])
dt.head()
```

	X	Y
0	-1.006572	0.324084

```

1 -2.276529 -0.385082
2 -0.704623 -0.676922
3 1.046060 0.611676

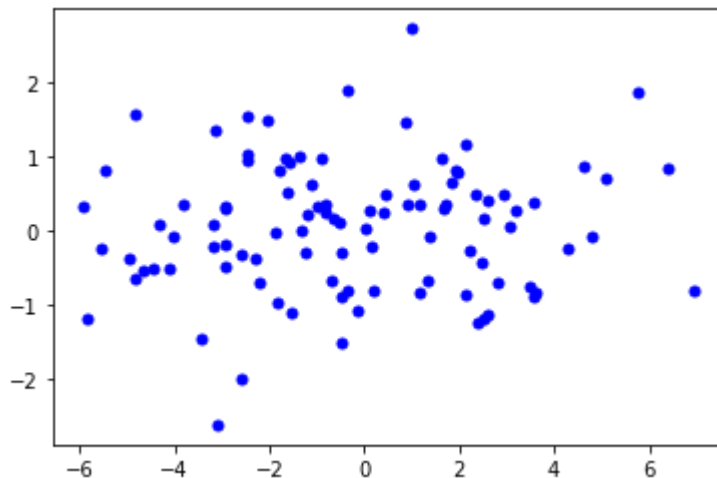
```

```

print("Generated Data : ")
plt.scatter(X, Y, s = 25, c = 'b')
plt.show()

```

Generated Data :



```

#2D Data Train and Test Split (80/20)
X_split = np.split(X, [80])
Y_split = np.split(Y, [80])

X_train = np.copy(X_split[0])
Y_train = np.copy(Y_split[0])

X_test = np.copy(X_split[1])
Y_test = np.copy(Y_split[1])

#data points 3D
n = 100

#draw from a Gaussian distribution

X0 = np.random.normal(loc=0.0, scale=3.0, size=int(n/4))
Y0 = np.random.normal(loc=-3.0, scale=1.0, size=int(n/4))
Z0 = np.random.normal(loc=-1.0, scale=1.0, size=int(n/4))

X1 = np.random.normal(loc=0.0, scale=3.0, size=int(n/4))
Y1 = np.random.normal(loc=1.0, scale=2.0, size=int(n/4))
Z1 = np.random.normal(loc=1.0, scale=1.0, size=int(n/4))

X2 = np.random.normal(loc=0.0, scale=3.0, size=int(n/4))
Y2 = np.random.normal(loc=3.0, scale=1.0, size=int(n/4))
Z2 = np.random.normal(loc=1.0, scale=1.0, size=int(n/4))

```

```
Z2 = np.random.normal(loc=4.0, scale=1.0, size=int(n/4))
```

```
X3 = np.random.normal(loc=0.0, scale=3.0, size=int(n/4))
```

```
Y3 = np.random.normal(loc=5.0, scale=3.0, size=int(n/4))
```

```
Z3 = np.random.normal(loc=-3.0, scale=1.0, size=int(n/4))
```

```
X = np.concatenate((X0, X1, X2, X3), axis=0)
```

```
Y = np.concatenate((Y0, Y1, Y2, Y3), axis=0)
```

```
Z = np.concatenate((Z0, Z1, Z2, Z3), axis=0)
```

```
dt = pd.DataFrame({'X':X, 'Y':Y, 'Z':Z}, columns=['X', 'Y', 'Z'])
```

```
dt.head()
```

	X	Y	Z
0	1.073362	-1.911049	-2.260884
1	1.682354	-2.935720	-0.082138
2	3.249154	-4.077745	1.122156
3	3.161406	-3.715304	0.032465
4	-4.133008	-2.320402	-2.519370

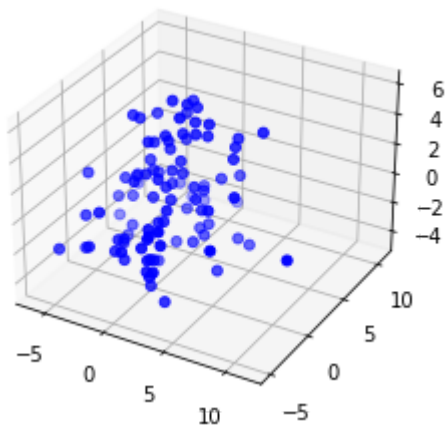
```
ax = plt.axes(projection = "3d")
```

```
print("Generated Data : ")
```

```
ax.scatter3D(X, Y, Z, s=25, c = "b")
```

```
plt.show()
```

Generated Data :



```
#3D Data Train and Test Split (80/20)
```

```
X_split = np.split(X, [80])
```

```
Y_split = np.split(Y, [80])
```

```
Z_split = np.split(Z, [80])
```

```
X_train = np.copy(X_split[0])  
Y_train = np.copy(Y_split[0])  
Z_train = np.copy(Z_split[0])  
  
X_test = np.copy(X_split[1])  
Y_test = np.copy(Y_split[1])  
Z_test = np.copy(Z_split[1])
```

---

✓ 0s completed at 4:44 PM

