

## Unsupervised Nearest Neighbor

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline

from sklearn.datasets import make_blobs
make_blobs()
```

```
[ -3.77635128, -0.27666407],
[ -3.28855378,  2.69695288],
[  2.34251141, -7.94662143],
[  1.04963145, -8.40245495],
[ -0.73861328,  6.43036542],
[ -0.23058955,  4.90883809],
[ -4.42765788,  1.81128433],
[  0.25257085, -9.08698419],
[ -0.75969127,  5.33220553],
[ -2.38442192,  1.1798099 ],
[  0.61196028, -6.9151369 ],
[ -0.67439449,  4.56499691],
[ -1.16958393, -8.52984717],
[ -3.97748158,  2.82847637],
[ -3.0887299 ,  2.04044633],
[ -4.5897851 ,  2.01841992],
[ -0.15829785,  4.00979403],
[ -2.35985865,  2.46877324],
[ -3.22570697,  1.67633468],
[ -1.76389097,  2.15469712],
[ -1.63806063,  3.27954565],
[ -3.88866745,  2.37943319],
[ -1.38834189,  5.5649749 ],
[ -2.8262003 ,  3.27089987],
[  0.86618075, -7.08258505],
[ -2.37945027,  3.19650235],
[ -1.36337023,  5.95574757],
[ -0.66454527, -8.10437524],
[ -3.1420014 ,  5.21909619],
[ -2.23223661,  6.19703507],
[ -0.94535774,  5.70789024],
[ -3.26335988,  1.91461378],
[  2.29462326, -9.37213544],
[ -2.61566251,  0.83784471],
[ -1.17321724,  2.84550193],
[ -2.9752284 ,  2.17738492],
[ -2.83497649,  3.54448869],
[  0.09153366,  6.00102105],
[  0.4602389 , -8.6393395 ],
[ -0.91802482,  6.82167257],
[ -2.87429647,  3.17804939],
[ -2.01615412,  5.50171623],
[  1.03849204, -8.88525937],
[  0.51218964, -8.85497215],
[ -3.77928547,  4.14641343],
```

```

    [-2.164951, 6.12184453],
    [-1.80525386, -9.28296267],
    [-1.77370694, 6.68058445],
    [1.13074092, -6.79376454],
    [-1.26687699, 2.99799732],
    [0.08945311, -7.82523851],
    [-3.16632273, 5.38393065],
    [-4.02147058, 2.36979739],
    [-0.31510686, -8.52954575]])],
array([[1, 1, 2, 0, 2, 1, 1, 0, 2, 2, 0, 1, 0, 0, 2, 1, 2, 1, 1, 1, 0, 0,
        0, 1, 1, 2, 2, 2, 2, 1, 1, 2, 0, 0, 1, 1, 0, 0, 1, 0, 0, 1, 0, 2,
        0, 1, 2, 2, 1, 1, 0, 0, 2, 1, 0, 2, 1, 0, 1, 2, 2, 2, 0, 2, 2, 0,
        0, 2, 0, 2, 1, 2, 0, 1, 0, 0, 0, 2, 1, 2, 2, 2, 2, 0, 1, 0, 2, 0,
        1, 1, 2, 0, 1, 0, 1, 2, 1, 0, 2, 1]))

```

```

X,_ = make_blobs(n_features=2, n_samples=10, cluster_std=5)
X,_

```

```

(array([[15.29965411, 3.82022821],
        [11.79083677, 4.52190852],
        [-8.81391305, 3.21335818],
        [8.96562697, 14.12213249],
        [-13.44445376, 3.85000191],
        [4.22695235, 9.95723082],
        [11.55921693, 6.68231567],
        [5.32280589, 6.27033793],
        [8.53691407, 14.32368728],
        [-3.65479195, -3.12447759]]), array([0, 0, 1, 2, 1, 2, 0, 0, 2, 1]))

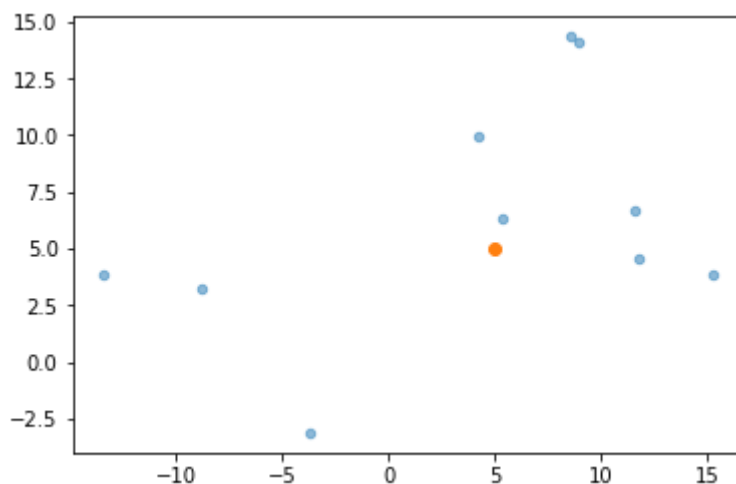
```

```

# X[:,0]
plt.scatter(X[:,0], X[:,1], s=20,alpha=.5)
plt.scatter([5],[5])

```

<matplotlib.collections.PathCollection at 0x7f47f593a610>



```

from sklearn.neighbors import NearestNeighbors
nn = NearestNeighbors(n_neighbors=2, algorithm='ball_tree')
nn.fit(X)

```

```
NearestNeighbors(algorithm='ball_tree', leaf_size=30, metric='minkowski',
                 metric_params=None, n_jobs=None, n_neighbors=2, p=2,
                 radius=1.0)
```

```
nn.kneighbors([[5,5]])
```

```
(array([[1.31071053, 5.01714461]]), array([[7, 5]]))
```

```
_,X_nearest = nn.kneighbors([[5,5]])
X_nearest
```

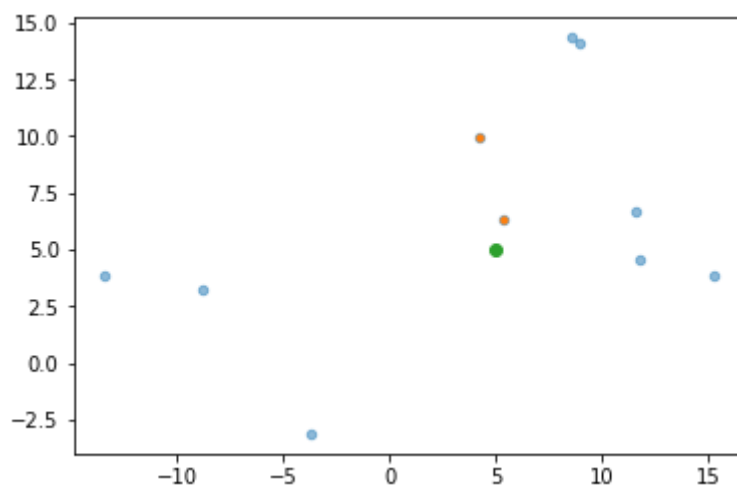
```
array([[7, 5]])
```

```
nearest = X[X_nearest[0]]
nearest
```

```
array([[5.32280589, 6.27033793],
       [4.22695235, 9.95723082]])
```

```
plt.scatter(X[:,0], X[:,1], s=20,alpha=.5)
plt.scatter(nearest[:,0], nearest[:,1], s=10 ,cmap='viridis')
plt.scatter([5],[5])
```

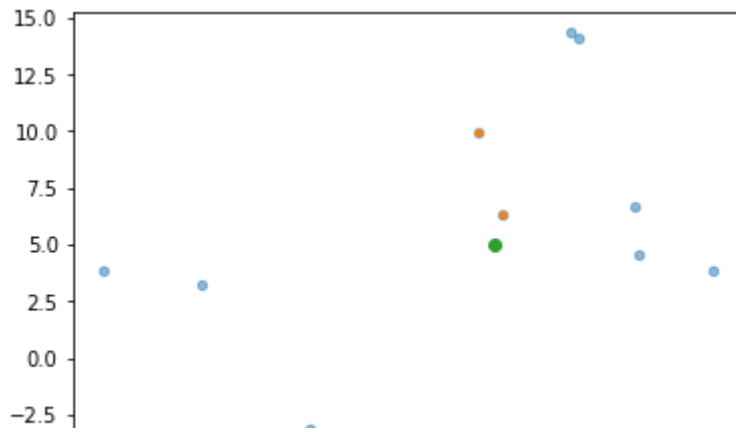
<matplotlib.collections.PathCollection at 0x7f47f29def90>



## Nearest Neighbors for Classification

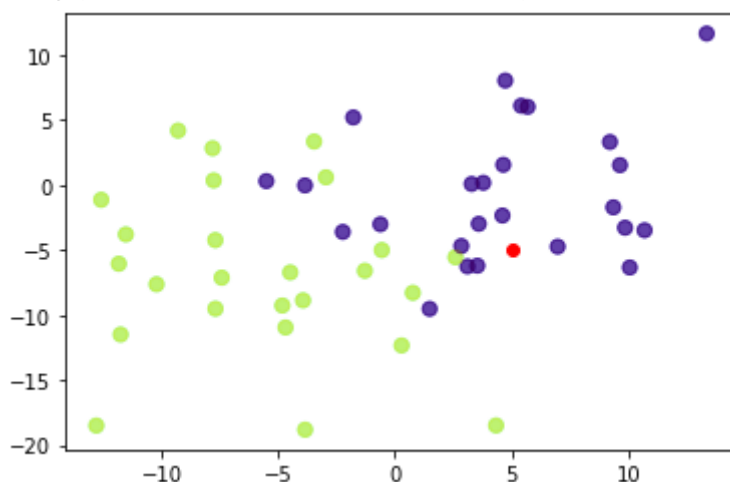
```
plt.scatter(X[:,0], X[:,1], s=20,alpha=.5)
plt.scatter(nearest[:,0], nearest[:,1], s=10 ,cmap='viridis')
plt.scatter([5],[5])
```

<matplotlib.collections.PathCollection at 0x7f47f29703d0>



```
from sklearn.neighbors import KNeighborsClassifier, RadiusNeighborsClassifier
knc = KNeighborsClassifier(n_neighbors=5)
rnc = RadiusNeighborsClassifier(radius=5)
X,Y = make_blobs(n_features=2, n_samples=50, cluster_std=5, centers=2)
Y
plt.scatter(X[:,0], X[:,1], s=50,alpha=.5, c=Y, cmap='winter')
plt.scatter([5],[-5],c='r')
plt.scatter(X[:,0], X[:,1], s=50,alpha=.5, c=Y)
knc.fit(X,Y)
rnc.fit(X,Y)
knc_pred = knc.predict([[5,-5]])
knc_pred
knc.kneighbors([[5,-5]])
_, knc_neighbors = knc.kneighbors([[5,-5]])
knc_neighbors
rnc_pred = rnc.predict([[5,-5]])
rnc_pred
rnc.radius_neighbors([[5,-5]])
_, rnc_neighbors = rnc.radius_neighbors([[5,-5]])
rnc_neighbors[0]
```

array([ 1, 8, 11, 18, 22, 27, 36])

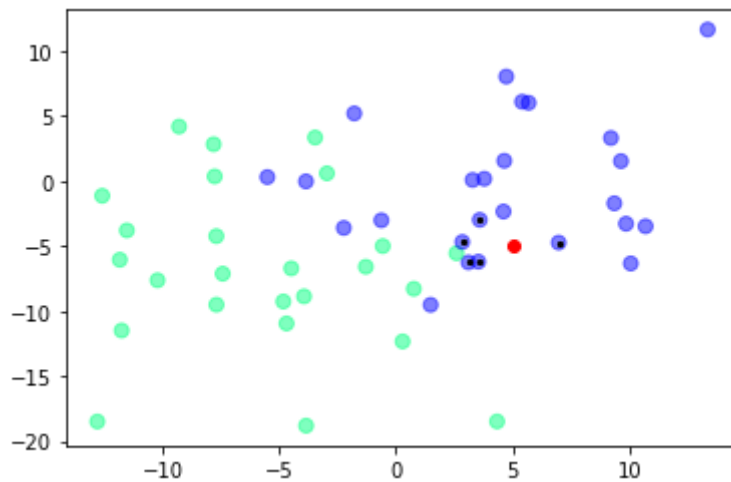


## Visualizing for NearestNeighbourClassifier

```
nearest = X[knc_neighbors[0]]
```

```
plt.scatter(X[:,0], X[:,1], s=50,alpha=.5, c=Y, cmap='winter')
plt.scatter([5],[-5],c='r')
plt.scatter(nearest[:,0], nearest[:,1], s=5, c='k')
```

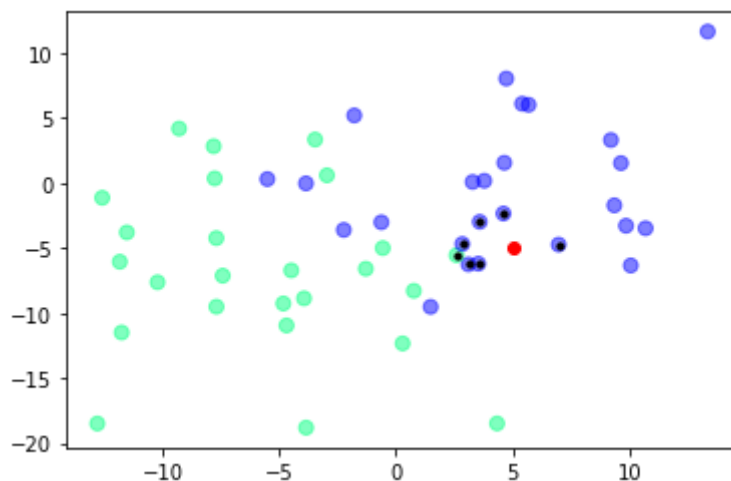
<matplotlib.collections.PathCollection at 0x7f47f2985290>



### Visualizing for RadiusNeighbourClassier

```
nearest = X[rnc_neighbors[0]]
plt.scatter(X[:,0], X[:,1], s=50,alpha=.5, c=Y, cmap='winter')
plt.scatter([5],[-5],c='r')
plt.scatter(nearest[:,0], nearest[:,1], s=10, c='k')
```

<matplotlib.collections.PathCollection at 0x7f47f286af90>



```
# rnc = RadiusNeighborsClassifier(radius=5)
rnc = RadiusNeighborsClassifier(radius=5, weights='distance')
rnc.fit(X,Y)
_, rnc_neighbors = rnc.radius_neighbors([[5,-5]])
rnc_neighbors

array([array([ 1,  8, 11, 18, 22, 27, 36])], dtype=object)
```

### Nearest Centroid Classier

```
from sklearn.neighbors.nearest_centroid import NearestCentroid
from sklearn.datasets import load_iris
iris = load_iris()
nc = NearestCentroid()
nc.fit(iris.data, iris.target)
nc.score(iris.data, iris.target)
```

```
↳ /usr/local/lib/python3.7/dist-packages/sklearn/utils/deprecation.py:144: FutureWarning:
    warnings.warn(message, FutureWarning)
0.9266666666666666
```

