Kevin Jairam

Week #4 Homework

Split the dataset used in this week's class into training_data (80%) and test_data (20%). Build kNN model with k=1, 15, 50 on the training set. Calculate the prediction accuracy on the test set.

In [1]:

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

In [3]:

```
url = "https://raw.githubusercontent.com/empathy87/The-Elements-of-Statistical
-Learning-Python-Notebooks/master/data/mixture.txt"
raw_data = pd.read_csv(url, sep=',')
raw_data.head()
```

Out[3]:

	x1	x2	y
0	2.526093	0.321050	0
1	0.366954	0.031462	0
2	0.768219	0.717486	0
3	0.693436	0.777194	0
4	-0.019837	0.867254	0

In [4]:

```
from sklearn.model_selection import train_test_split
training_data, test_data = train_test_split(raw_data, test_size=0.2)
```

In [7]:

```
#training_data for 80%
training_data.shape
```

Out[7]:

```
(160, 3)
```

In [9]:

```
#test_data for 20%
test_data.shape
```

Out[9]:

```
(40, 3)
```

KNN For k = 1

In [24]:

```
from sklearn.neighbors import KNeighborsClassifier
knn1 = KNeighborsClassifier(n_neighbors=1)
knn1.fit(training_data[["x1", "x2"]], training_data[["y"]])
```

```
C:\Users\Kevin J\Anaconda3\lib\site-packages\ipykernel_launcher.py:3: DataC
onversionWarning: A column-vector y was passed when a 1d array was expecte
d. Please change the shape of y to (n_samples, ), for example using ravel
().
  This is separate from the ipykernel package so we can avoid doing imports
until
```

Out[24]:

```
KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
                     metric_params=None, n_jobs=None, n_neighbors=1, p=2,
                     weights='uniform')
```

In [25]:

```
predictions = knn1.predict(test_data[["x1", "x2"]])
predictions
```

Out[25]:

```
array([0, 0, 1, 0, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1,
       0, 1, 1, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0], dtype=int64)
```

In [26]:

```
from sklearn.metrics import accuracy_score
accuracy_score(test_data['y'], predictions)
```

Out[26]:

```
0.75
```

KNN For k = 15

In [27]:

```
knn15 = KNeighborsClassifier(n_neighbors=15)
knn15.fit(training_data[["x1", "x2"]], training_data[["y"]])
```

```
C:\Users\Kevin J\Anaconda3\lib\site-packages\ipykernel_launcher.py:2: DataC
onversionWarning: A column-vector y was passed when a 1d array was expecte
d. Please change the shape of y to (n_samples, ), for example using ravel
().
```

Out[27]:

```
KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
                     metric_params=None, n_jobs=None, n_neighbors=15, p=2,
                     weights='uniform')
```

In [29]:

```
predictions1 = knn15.predict(test_data[["x1", "x2"]])
predictions1
```

Out[29]:

```
array([0, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 0, 0, 0, 1, 1, 1,
       0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 0, 1, 1, 1, 0, 0], dtype=int64)
```

In [30]:

```
accuracy_score(test_data['y'], predictions1)
```

Out[30]:

```
0.75
```

KNN For k = 50

In [37]:

```
from sklearn.neighbors import KNeighborsClassifier
knn5 = KNeighborsClassifier(n_neighbors=50)
knn5.fit(training_data[["x1", "x2"]], training_data[["y"]])
```

```
C:\Users\Kevin J\Anaconda3\lib\site-packages\ipykernel_launcher.py:3: DataC
onversionWarning: A column-vector y was passed when a 1d array was expecte
d. Please change the shape of y to (n_samples, ), for example using ravel
().
  This is separate from the ipykernel package so we can avoid doing imports
until
```

Out[37]:

```
KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
                     metric_params=None, n_jobs=None, n_neighbors=50, p=2,
                     weights='uniform')
```

In [38]:

```
predictions2 = knn5.predict(test_data[["x1", "x2"]])
predictions2
```

Out[38]:

```
array([0, 0, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1,
       0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0], dtype=int64)
```