**Assignment 2 ( KNN)**

from math import \*

def euclidean\_distance(row1, row2):

distance = 0.0

for i in range(len(row1)-1):

distance += (row1[i] - row2[i])\*\*2

return sqrt(distance)

def get\_neighbors(train, test\_row, num\_neighbors):

distances = list()

for train\_row in train:

dist = euclidean\_distance(train\_row, test\_row)

distances.append((train\_row, dist))

distances.sort(key=lambda tup: tup[1])

neighbors = list()

for i in range(num\_neighbors):

neighbors.append(distances[i][0])

return neighbors

train\_data = [[2, 4, "Orange"], [4, 4, "Blue"], [4, 6, "Orange"], [4, 2, "Orange"], [6, 2, "Blue"], [6, 4, "Orange"]]

test\_data = [[6, 6]]

for item in test\_data:

print("Data point : ", item)

neighbors = get\_neighbors(train\_data, item, 3)

print("Nearest Neighbors : ", neighbors)

output\_values = [row[-1] for row in neighbors]

prediction = max(set(output\_values), key=output\_values.count)

print("Prediction : ", prediction, "\n")

**Output:**

Data point : [6, 6]

Nearest Neighbors : [[4, 6, 'Orange'], [6, 4, 'Orange'], [4, 4, 'Blue']]

Prediction : Orange