Slddhant Barua Lab 3 Report

```
from PIL import Image
import numpy as np
import math as mt
import operator
# from sklearn.neighbors import KNeighborsClassifier

k = 1


def Train(bin_size):
   # there are 12 images that are used to train  the classifier , now each of these
12 Training_Set(s) will have an
   # assigned binspace and then we multiply that by 3 because there are 3 color
channels i.e. RGB
   model = [['Training_Set_1', [0 for i in range(bin_size * 3)]],
            ['Training_Set_1', [0 for i in range(bin_size * 3)]],
            ['Training_Set_1', [0 for i in range(bin_size * 3)]],
            ['Training_Set_1', [0 for i in range(bin_size * 3)]],
            ['Training_Set_2', [0 for i in range(bin_size * 3)]],
            ['Training_Set_2', [0 for i in range(bin_size * 3)]],
            ['Training_Set_2', [0 for i in range(bin_size * 3)]],
            ['Training_Set_2', [0 for i in range(bin_size * 3)]],
            ['Training_Set_3', [0 for i in range(bin_size * 3)]],
            ['Training_Set_3', [0 for i in range(bin_size * 3)]],
            ['Training_Set_3', [0 for i in range(bin_size * 3)]],
            ['Training_Set_3', [0 for i in range(bin_size * 3)]]]
   index = 0
   for i in range(0, 12):
       if i == 0:
           input = Image.open("./ImClass/Train/coast_train1.jpg")
           c_label = "coast"
           file = "coast_train1"
       if i == 1:
           input = Image.open("./ImClass/Train/coast_train2.jpg")
           c_label = "coast"
           file = "coast_train2"
       if i == 2:
           input = Image.open("./ImClass/Train/coast_train3.jpg")
           c_label = "coast"
           file = "coast_train3"
       if i == 3:
           input = Image.open("./ImClass/Train/coast_train4.jpg")
           c_label = "coast"
           file = "coast_train4"
       if i == 4:
           input = Image.open("./ImClass/Train/forest_train1.jpg")
           c_label = "forest"
           file = "forest_train1"
```

```
        if i == 5:
            input = Image.open("./ImClass/Train/forest_train2.jpg")
            c_label = "forest"
            file = "forest_train2"
        if i == 6:
            input = Image.open("./ImClass/Train/forest_train3.jpg")
            c_label = "forest"
            file = "forest_train3"
        if i == 7:
            input = Image.open("./ImClass/Train/forest_train4.jpg")
            c_label = "forest"
            file = "forest_train4"
        if i == 8:
            input = Image.open("./ImClass/Train/insidecity_train1.jpg")
            c_label = "insidecity"
            file = "insidecity_train1"
        if i == 9:
            input = Image.open("./ImClass/Train/insidecity_train2.jpg")
            c_label = "insidecity"
            file = "insidecity_train2"
        if i == 10:
            input = Image.open("./ImClass/Train/insidecity_train3.jpg")
            c_label = "insidecity"
            file = "insidecity_train3"
        if i == 11:
            input = Image.open("./ImClass/Train/insidecity_train4.jpg")
            c_label = "insidecity"
            file = "insidecity_train4"

        ip_image = np.asarray(input)
        r, c, _ = ip_image.shape
        model[index][0] = c_label  # We assign the index for that particular model
i.e. label them forest , coast , inCity
        div_val = (256 / bin_size)
        for i in range(r):
            for j in range(c):
                r_val = int((ip_image[i][j][0]) / div_val)
                g_val = int((ip_image[i][j][1]) / div_val)
                b_val = int((ip_image[i][j][2]) / div_val)
                model[index][1][r_val] = model[index][1][r_val] + 1
                model[index][1][g_val + bin_size] = model[index][1][g_val + bin_size]
+ 1
                model[index][1][b_val + bin_size + bin_size] = model[index][1][b_val
+ bin_size + bin_size] + 1

        count = 0
        for iCount in model[index][1]:
            count = count + iCount
        if count / 3 == (r * c):
            print('The correct histogram for ' + file + ' has been generated.')
```

```
        else:
            print('The histogram for ' + file + 'is not correctly generated.')
        index = index + 1

print("**************************************************************************
************************")
    return model


def Test(trained_model):
    bin_size = int(len(trained_model[0][1]) / 3)
    predict_true = 0
    predict_false = 0

    count = 0

    for i in range(0, 12):
        if i == 0:
            input = Image.open("./ImClass/Test/coast_test1.jpg")
            initial_label = "coast"
            file = "coast_test1"
        if i == 1:
            input = Image.open("./ImClass/Test/coast_test2.jpg")
            initial_label = "coast"
            file = "coast_test2"
        if i == 2:
            input = Image.open("./ImClass/Test/coast_test3.jpg")
            initial_label = "coast"
            file = "coast_test3"
        if i == 3:
            input = Image.open("./ImClass/Test/coast_test4.jpg")
            initial_label = "coast"
            file = "coast_test4"
        if i == 4:
            input = Image.open("./ImClass/Test/forest_test1.jpg")
            initial_label = "forest"
            file = "forest_test1"
        if i == 5:
            input = Image.open("./ImClass/Test/forest_test2.jpg")
            initial_label = "forest"
            file = "forest_test2"
        if i == 6:
            input = Image.open("./ImClass/Test/forest_test3.jpg")
            initial_label = "forest"
            file = "forest_test3"
        if i == 7:
            input = Image.open("./ImClass/Test/forest_test4.jpg")
            initial_label = "forest"
            file = "forest_test4"
        if i == 8:
```

```
            input = Image.open("./ImClass/Test/insidecity_test1.jpg")
            initial_label = "insidecity"
            file = "insidecity_test1"
        if i == 9:
            input = Image.open("./ImClass/Test/insidecity_test2.jpg")
            initial_label = "insidecity"
            file = "insidecity_test2"
        if i == 10:
            input = Image.open("./ImClass/Test/insidecity_test3.jpg")
            initial_label = "insidecity"
            file = "insidecity_test3"
        if i == 11:
            input = Image.open("./ImClass/Test/insidecity_test4.jpg")
            initial_label = "insidecity"
            file = "insidecity_test4"

        print(trained_model)
        ip_image = np.asarray(input)
        test_model = [0 for i in range(bin_size * 3)]
        r, c, _ = ip_image.shape
        div_val = (256 / bin_size)
        for i in range(r):
            for j in range(c):
                r_val = int((ip_image[i][j][0]) / div_val)
                g_val = int((ip_image[i][j][1]) / div_val)
                b_val = int((ip_image[i][j][2]) / div_val)
                test_model[r_val] = test_model[r_val] + 1
                test_model[g_val + bin_size] = test_model[g_val + bin_size] + 1
                test_model[b_val + bin_size + bin_size] = test_model[b_val + bin_size
+ bin_size] + 1
        threshold = 999999.0
        # src:
https://machinelearningmastery.com/tutorial-to-implement-k-nearest-neighbors-in-pyth
on-from-scratch/
        distances = []
        for mod in trained_model:
            distance = 0.0
            for i in range(len(mod[1])):
                # for j in range(len(mod[1])):
                #     for k in range(len(mod[1])):
                distance = distance + (((mod[1][i] - test_model[i]) ** 2))  # this
calculates the euclidean distance
                # + ((mod[1][j] - test_model[j]) ** 2) + (
                #         (mod[1][k] - test_model[k]) ** 2))
            distances.append((mod[0], distance))
            if (mt.sqrt(distance)) < threshold:  # for printing purposes
                # print(threshold)
                threshold = mt.sqrt(distance)
                test_label = mod[0]
```

```
        distances.sort(key=operator.itemgetter(1))  # sorting the distances
        # print(distances)
        neighbors = []
        for x in range(k):  # this is the value of k
            neighbors.append(distances[x][0]) #appends 3 neighbours which are closest
based on the sort function

        # BLOCK 1
        ######################################################################
        if initial_label == test_label:
            predict_true = predict_true + 1
        else:
            predict_false = predict_false + 1

        # BLOCK 2  (Note: Uncomment block 2 and comment block 1 if you want to run
tests where K value is greater than 1 )
        ######################################################################
        # if initial_label in neighbors:
        #     count += 1
        # else:
        #     count = 0
        #
        # if count > 2:
        #     predict_true = predict_true + 1
        # else:
        #     predict_false = predict_false + 1
        print('The image: ' + file + ', assigned the class: ' + initial_label + ', is
assigned: ', test_label,
              ', test_label.')

print("================================================================================
======================")
    accuracy = (predict_true / (predict_true + predict_false)) * 100
    print(
        '\n Accuracy of the image classifier: ' + str(accuracy) + ', number of bins:
' + str(bin_size) + ', k value: ',
        k, '.')

print("================================================================================
======================")


def main():
    bin_size = 4
    Train_Model = Train(bin_size)
    Test(Train_Model)


main()
```

We have the following observations
Output when bin size is 8 and we have k = 1 or the one nearest neighbour

TRAIN Function O/P

```
The correct histogram for coast_train1 has been generated.
********************************************************************************************************************
The correct histogram for coast_train2 has been generated.
********************************************************************************************************************
The correct histogram for coast_train3 has been generated.
********************************************************************************************************************
The correct histogram for coast_train4 has been generated.
********************************************************************************************************************
The correct histogram for forest_train1 has been generated.
********************************************************************************************************************
The correct histogram for forest_train2 has been generated.
********************************************************************************************************************
The correct histogram for forest_train3 has been generated.
********************************************************************************************************************
The correct histogram for forest_train4 has been generated.
********************************************************************************************************************
The correct histogram for insidecity_train1 has been generated.
********************************************************************************************************************
The correct histogram for insidecity_train2 has been generated.
********************************************************************************************************************
The correct histogram for insidecity_train3 has been generated.
********************************************************************************************************************
The correct histogram for insidecity_train4 has been generated.
********************************************************************************************************************
```

TEST Function O/P

```
**********************************************************************************************************
[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096,
The image: coast_test1, assigned the class: coast, is assigned:  coast , test_label.
==========================================================================================================
[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096,
The image: coast_test2, assigned the class: coast, is assigned:  coast , test_label.
==========================================================================================================
[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096,
The image: coast_test3, assigned the class: coast, is assigned:  coast , test_label.
==========================================================================================================
[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096,
The image: coast_test4, assigned the class: coast, is assigned:  coast , test_label.
==========================================================================================================
[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096,
The image: forest_test1, assigned the class: forest, is assigned:  forest , test_label.
==========================================================================================================
[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096,
The image: forest_test2, assigned the class: forest, is assigned:  forest , test_label.
==========================================================================================================
[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096,
The image: forest_test3, assigned the class: forest, is assigned:  forest , test_label.
==========================================================================================================
```

```
==========================================================================================================
[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 1
The image: forest_test4, assigned the class: forest, is assigned:  forest , test_label.
==========================================================================================================
[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 1
The image: insidecity_test1, assigned the class: insidecity, is assigned:  forest , test_label.
==========================================================================================================
[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 1
The image: insidecity_test2, assigned the class: insidecity, is assigned:  forest , test_label.
==========================================================================================================
[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 1
The image: insidecity_test3, assigned the class: insidecity, is assigned:  insidecity , test_label.
==========================================================================================================
[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 1
The image: insidecity_test4, assigned the class: insidecity, is assigned:  insidecity , test_label.
==========================================================================================================

 Accuracy of the image classifier: 83.33333333333334, number of bins: 8, k value:  1 .
==========================================================================================================
```

Output when bin size is 4 and we have k = 1 or the one nearest neighbour

TRAIN O/P

```
The correct histogram for coast_train1 has been generated.
********************************************************************************************************
The correct histogram for coast_train2 has been generated.
********************************************************************************************************
The correct histogram for coast_train3 has been generated.
********************************************************************************************************
The correct histogram for coast_train4 has been generated.
********************************************************************************************************
The correct histogram for forest_train1 has been generated.
********************************************************************************************************
The correct histogram for forest_train2 has been generated.
********************************************************************************************************
The correct histogram for forest_train3 has been generated.
********************************************************************************************************
The correct histogram for forest_train4 has been generated.
********************************************************************************************************
The correct histogram for insidecity_train1 has been generated.
********************************************************************************************************
The correct histogram for insidecity_train2 has been generated.
********************************************************************************************************
The correct histogram for insidecity_train3 has been generated.
********************************************************************************************************
The correct histogram for insidecity_train4 has been generated.
********************************************************************************************************
```

TEST O/P

```
================================================================================================
[['coast', [20971, 17267, 20213, 7085, 19709, 14173, 25125, 6529, 17507, 8341, 23537, 16151]], ['coast', [33710, 15710, 7775, 8341, 4815, 22478, 27124, 11
The image: coast_test2, assigned the class: coast, is assigned:  coast , test_label.
================================================================================================
[['coast', [20971, 17267, 20213, 7085, 19709, 14173, 25125, 6529, 17507, 8341, 23537, 16151]], ['coast', [33710, 15710, 7775, 8341, 4815, 22478, 27124, 11
The image: coast_test3, assigned the class: coast, is assigned:  coast , test_label.
================================================================================================
[['coast', [20971, 17267, 20213, 7085, 19709, 14173, 25125, 6529, 17507, 8341, 23537, 16151]], ['coast', [33710, 15710, 7775, 8341, 4815, 22478, 27124, 11
The image: coast_test4, assigned the class: coast, is assigned:  coast , test_label.
================================================================================================
[['coast', [20971, 17267, 20213, 7085, 19709, 14173, 25125, 6529, 17507, 8341, 23537, 16151]], ['coast', [33710, 15710, 7775, 8341, 4815, 22478, 27124, 11
The image: forest_test1, assigned the class: forest, is assigned:  forest , test_label.
================================================================================================
[['coast', [20971, 17267, 20213, 7085, 19709, 14173, 25125, 6529, 17507, 8341, 23537, 16151]], ['coast', [33710, 15710, 7775, 8341, 4815, 22478, 27124, 11
The image: forest_test2, assigned the class: forest, is assigned:  forest , test_label.
================================================================================================
[['coast', [20971, 17267, 20213, 7085, 19709, 14173, 25125, 6529, 17507, 8341, 23537, 16151]], ['coast', [33710, 15710, 7775, 8341, 4815, 22478, 27124, 11
The image: forest_test3, assigned the class: forest, is assigned:  forest , test_label.
================================================================================================
[['coast', [20971, 17267, 20213, 7085, 19709, 14173, 25125, 6529, 17507, 8341, 23537, 16151]], ['coast', [33710, 15710, 7775, 8341, 4815, 22478, 27124, 11
The image: forest_test4, assigned the class: forest, is assigned:  forest , test_label.
================================================================================================
[['coast', [20971, 17267, 20213, 7085, 19709, 14173, 25125, 6529, 17507, 8341, 23537, 16151]], ['coast', [33710, 15710, 7775, 8341, 4815, 22478, 27124, 11
The image: insidecity_test1, assigned the class: insidecity, is assigned:  forest , test_label.
================================================================================================
```

```
==========================================================================================
[['coast', [20971, 17267, 20213, 7085, 19709, 14173, 25125, 6529, 17507, 8341, 23537, 16151]], ['coast', [33710, 15710, 7775, 8341, 4815, 22478, 271
The image: insidecity_test2, assigned the class: insidecity, is assigned:  insidecity , test_label.
==========================================================================================
[['coast', [20971, 17267, 20213, 7085, 19709, 14173, 25125, 6529, 17507, 8341, 23537, 16151]], ['coast', [33710, 15710, 7775, 8341, 4815, 22478, 271
The image: insidecity_test3, assigned the class: insidecity, is assigned:  insidecity , test_label.
==========================================================================================
[['coast', [20971, 17267, 20213, 7085, 19709, 14173, 25125, 6529, 17507, 8341, 23537, 16151]], ['coast', [33710, 15710, 7775, 8341, 4815, 22478, 271
The image: insidecity_test4, assigned the class: insidecity, is assigned:  coast , test_label.
==========================================================================================

 Accuracy of the image classifier: 83.33333333333334, number of bins: 4, k value:  1 .
==========================================================================================
```

Output when bin size is 16 and we have k = 1 or the one nearest neighbour

TRAIN O/P

TEST O/P

```
[['coast', [13895, 2245, 2410, 2421, 3097, 3923, 5042, 5205, 6051, 8381, 5150, 631, 558, 612, 629, 5286, 12172, 2906, 2095, 2536, 2680, 2128, 3288, 6077, 6245, 9162, 9078, 64
The image: coast_test1, assigned the class: coast, is assigned:  coast , test_label.
================================================================================================
[['coast', [13895, 2245, 2410, 2421, 3097, 3923, 5042, 5205, 6051, 8381, 5150, 631, 558, 612, 629, 5286, 12172, 2906, 2095, 2536, 2680, 2128, 3288, 6077, 6245, 9162, 9078, 64
The image: coast_test2, assigned the class: coast, is assigned:  coast , test_label.
================================================================================================
[['coast', [13895, 2245, 2410, 2421, 3097, 3923, 5042, 5205, 6051, 8381, 5150, 631, 558, 612, 629, 5286, 12172, 2906, 2095, 2536, 2680, 2128, 3288, 6077, 6245, 9162, 9078, 64
The image: coast_test3, assigned the class: coast, is assigned:  coast , test_label.
================================================================================================
[['coast', [13895, 2245, 2410, 2421, 3097, 3923, 5042, 5205, 6051, 8381, 5150, 631, 558, 612, 629, 5286, 12172, 2906, 2095, 2536, 2680, 2128, 3288, 6077, 6245, 9162, 9078, 64
The image: coast_test4, assigned the class: coast, is assigned:  coast , test_label.
================================================================================================
[['coast', [13895, 2245, 2410, 2421, 3097, 3923, 5042, 5205, 6051, 8381, 5150, 631, 558, 612, 629, 5286, 12172, 2906, 2095, 2536, 2680, 2128, 3288, 6077, 6245, 9162, 9078, 64
The image: forest_test1, assigned the class: forest, is assigned:  forest , test_label.
================================================================================================
[['coast', [13895, 2245, 2410, 2421, 3097, 3923, 5042, 5205, 6051, 8381, 5150, 631, 558, 612, 629, 5286, 12172, 2906, 2095, 2536, 2680, 2128, 3288, 6077, 6245, 9162, 9078, 64
The image: forest_test2, assigned the class: forest, is assigned:  forest , test_label.
================================================================================================
[['coast', [13895, 2245, 2410, 2421, 3097, 3923, 5042, 5205, 6051, 8381, 5150, 631, 558, 612, 629, 5286, 12172, 2906, 2095, 2536, 2680, 2128, 3288, 6077, 6245, 9162, 9078, 64
The image: forest_test3, assigned the class: forest, is assigned:  forest , test_label.
================================================================================================
[['coast', [13895, 2245, 2410, 2421, 3097, 3923, 5042, 5205, 6051, 8381, 5150, 631, 558, 612, 629, 5286, 12172, 2906, 2095, 2536, 2680, 2128, 3288, 6077, 6245, 9162, 9078, 64
The image: forest_test4, assigned the class: forest, is assigned:  forest , test_label.
================================================================================================
```

```
    ================================================================================================
    [['coast', [13895, 2245, 2410, 2421, 3097, 3923, 5042, 5205, 6051, 8381, 5150, 631, 558, 612, 629, 5286, 12172, 2906, 2095, 2536, 2680, 2128, 3288, 6077, 6245, 9162, 90
    The image: insidecity_test1, assigned the class: insidecity, is assigned:  forest , test_label.
    ================================================================================================
    [['coast', [13895, 2245, 2410, 2421, 3097, 3923, 5042, 5205, 6051, 8381, 5150, 631, 558, 612, 629, 5286, 12172, 2906, 2095, 2536, 2680, 2128, 3288, 6077, 6245, 9162, 90
    The image: insidecity_test2, assigned the class: insidecity, is assigned:  forest , test_label.
    ================================================================================================
    [['coast', [13895, 2245, 2410, 2421, 3097, 3923, 5042, 5205, 6051, 8381, 5150, 631, 558, 612, 629, 5286, 12172, 2906, 2095, 2536, 2680, 2128, 3288, 6077, 6245, 9162, 90
    The image: insidecity_test3, assigned the class: insidecity, is assigned:  forest , test_label.
    ================================================================================================
    [['coast', [13895, 2245, 2410, 2421, 3097, 3923, 5042, 5205, 6051, 8381, 5150, 631, 558, 612, 629, 5286, 12172, 2906, 2095, 2536, 2680, 2128, 3288, 6077, 6245, 9162, 90
    The image: insidecity_test4, assigned the class: insidecity, is assigned:  insidecity , test_label.
    ================================================================================================


     Accuracy of the image classifier: 75.0, number of bins: 16, k value:  1 .
    ================================================================================================
```

Output when bin size is 32 and we have k = 1 or the one nearest neighbour

TRAIN O/P

```
The correct histogram for coast_train1 has been generated.
********************************************************************************************************************
The correct histogram for coast_train2 has been generated.
********************************************************************************************************************
The correct histogram for coast_train3 has been generated.
********************************************************************************************************************
The correct histogram for coast_train4 has been generated.
********************************************************************************************************************
The correct histogram for forest_train1 has been generated.
********************************************************************************************************************
The correct histogram for forest_train2 has been generated.
********************************************************************************************************************
The correct histogram for forest_train3 has been generated.
********************************************************************************************************************
The correct histogram for forest_train4 has been generated.
********************************************************************************************************************
The correct histogram for insidecity_train1 has been generated.
********************************************************************************************************************
The correct histogram for insidecity_train2 has been generated.
********************************************************************************************************************
The correct histogram for insidecity_train3 has been generated.
********************************************************************************************************************
The correct histogram for insidecity_train4 has been generated.
********************************************************************************************************************
```

TEST O/P

```
[['coast', [801, 13094, 1195, 1050, 1172, 1238, 1207, 1214, 1286, 1811, 1831, 2092, 2484, 2558, 2693, 2512, 2713, 3338, 3685, 4696, 3573, 1577, 329, 302, 300, 258
The image: coast_test1, assigned the class: coast, is assigned:  coast , test_label.
============================================================================================
[['coast', [801, 13094, 1195, 1050, 1172, 1238, 1207, 1214, 1286, 1811, 1831, 2092, 2484, 2558, 2693, 2512, 2713, 3338, 3685, 4696, 3573, 1577, 329, 302, 300, 258
The image: coast_test2, assigned the class: coast, is assigned:  coast , test_label.
============================================================================================
[['coast', [801, 13094, 1195, 1050, 1172, 1238, 1207, 1214, 1286, 1811, 1831, 2092, 2484, 2558, 2693, 2512, 2713, 3338, 3685, 4696, 3573, 1577, 329, 302, 300, 258
The image: coast_test3, assigned the class: coast, is assigned:  coast , test_label.
============================================================================================
[['coast', [801, 13094, 1195, 1050, 1172, 1238, 1207, 1214, 1286, 1811, 1831, 2092, 2484, 2558, 2693, 2512, 2713, 3338, 3685, 4696, 3573, 1577, 329, 302, 300, 258
The image: coast_test4, assigned the class: coast, is assigned:  coast , test_label.
============================================================================================
[['coast', [801, 13094, 1195, 1050, 1172, 1238, 1207, 1214, 1286, 1811, 1831, 2092, 2484, 2558, 2693, 2512, 2713, 3338, 3685, 4696, 3573, 1577, 329, 302, 300, 258
The image: forest_test1, assigned the class: forest, is assigned:  forest , test_label.
============================================================================================
[['coast', [801, 13094, 1195, 1050, 1172, 1238, 1207, 1214, 1286, 1811, 1831, 2092, 2484, 2558, 2693, 2512, 2713, 3338, 3685, 4696, 3573, 1577, 329, 302, 300, 258
The image: forest_test2, assigned the class: forest, is assigned:  forest , test_label.
============================================================================================
[['coast', [801, 13094, 1195, 1050, 1172, 1238, 1207, 1214, 1286, 1811, 1831, 2092, 2484, 2558, 2693, 2512, 2713, 3338, 3685, 4696, 3573, 1577, 329, 302, 300, 258
The image: forest_test3, assigned the class: forest, is assigned:  forest , test_label.
============================================================================================
[['coast', [801, 13094, 1195, 1050, 1172, 1238, 1207, 1214, 1286, 1811, 1831, 2092, 2484, 2558, 2693, 2512, 2713, 3338, 3685, 4696, 3573, 1577, 329, 302, 300, 258
The image: forest_test4, assigned the class: forest, is assigned:  forest , test_label.
============================================================================================
```

```
[['coast', [801, 13094, 1195, 1050, 1172, 1238, 1207, 1214, 1286, 1811, 1831, 2092, 2484, 2558, 2693, 2512, 2713, 3338, 3685, 4696, 3573, 1577, 329, 302,
The image: insidecity_test1, assigned the class: insidecity, is assigned:  forest , test_label.
==========
[['coast', [801, 13094, 1195, 1050, 1172, 1238, 1207, 1214, 1286, 1811, 1831, 2092, 2484, 2558, 2693, 2512, 2713, 3338, 3685, 4696, 3573, 1577, 329, 302,
The image: insidecity_test2, assigned the class: insidecity, is assigned:  forest , test_label.
==========
[['coast', [801, 13094, 1195, 1050, 1172, 1238, 1207, 1214, 1286, 1811, 1831, 2092, 2484, 2558, 2693, 2512, 2713, 3338, 3685, 4696, 3573, 1577, 329, 302,
The image: insidecity_test3, assigned the class: insidecity, is assigned:  forest , test_label.
==========
[['coast', [801, 13094, 1195, 1050, 1172, 1238, 1207, 1214, 1286, 1811, 1831, 2092, 2484, 2558, 2693, 2512, 2713, 3338, 3685, 4696, 3573, 1577, 329, 302,
The image: insidecity_test4, assigned the class: insidecity, is assigned:  insidecity , test_label.
==========

 Accuracy of the image classifier: 75.0, number of bins: 32, k value:  1 .
==========
```

When we want to test 3 nearest neighbours change the “k” value of code form 1 to 3 And then comment Block 1 i.e. lines 196 -199 and then uncomment Block 2 i.e. lines 202 - 211
We see the following outputs for the 3 nearest neighbours classification scheme with bin size = 8

TRAIN O/P

```
The correct histogram for coast_train1 has been generated.
**********
The correct histogram for coast_train2 has been generated.
**********
The correct histogram for coast_train3 has been generated.
**********
The correct histogram for coast_train4 has been generated.
**********
The correct histogram for forest_train1 has been generated.
**********
The correct histogram for forest_train2 has been generated.
**********
The correct histogram for forest_train3 has been generated.
**********
The correct histogram for forest_train4 has been generated.
**********
The correct histogram for insidecity_train1 has been generated.
**********
The correct histogram for insidecity_train2 has been generated.
**********
The correct histogram for insidecity_train3 has been generated.
**********
The correct histogram for insidecity_train4 has been generated.
**********
```

TEST O/P

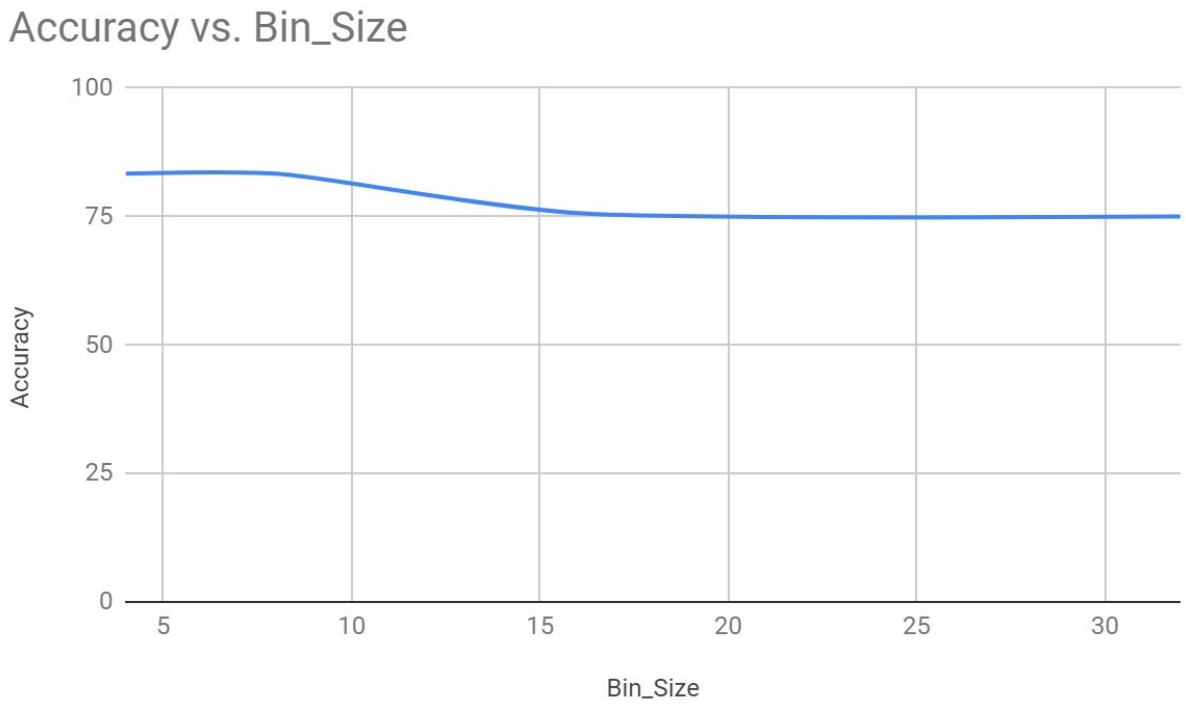
```
[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096, 5017, 3324,
The image: coast_test1, assigned the class: coast, is assigned:  coast , test_label.
==========================================================================================

[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096, 5017, 3324,
The image: coast_test2, assigned the class: coast, is assigned:  coast , test_label.
==========================================================================================

[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096, 5017, 3324,
The image: coast_test3, assigned the class: coast, is assigned:  coast , test_label.
==========================================================================================

[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096, 5017, 3324,
The image: coast_test4, assigned the class: coast, is assigned:  coast , test_label.
==========================================================================================

[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096, 5017, 3324,
The image: forest_test1, assigned the class: forest, is assigned:  forest , test_label.
==========================================================================================

[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096, 5017, 3324,
The image: forest_test2, assigned the class: forest, is assigned:  forest , test_label.
==========================================================================================

[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096, 5017, 3324,
The image: forest_test3, assigned the class: forest, is assigned:  forest , test_label.
==========================================================================================

[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096, 5017, 3324,
The image: forest_test4, assigned the class: forest, is assigned:  forest , test_label.
==========================================================================================
```

```
[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096, 5017, 3324,
The image: insidecity_test1, assigned the class: insidecity, is assigned:  forest , test_label.
==========================================================================================

[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096, 5017, 3324,
The image: insidecity_test2, assigned the class: insidecity, is assigned:  forest , test_label.
==========================================================================================

[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096, 5017, 3324,
The image: insidecity_test3, assigned the class: insidecity, is assigned:  insidecity , test_label.
==========================================================================================

[['coast', [16140, 4831, 7020, 10247, 14432, 5781, 1170, 5915, 15078, 4631, 4808, 9365, 15407, 9718, 1224, 5305, 14411, 3096, 5017, 3324,
The image: insidecity_test4, assigned the class: insidecity, is assigned:  insidecity , test_label.
==========================================================================================


 Accuracy of the image classifier: 83.33333333333334, number of bins: 8, k value:  3 .
==========================================================================================
```

We notice that the accuracy of the predictions is almost identical despite the value of k being = 3 i.e. using 3 nearest neighbours instead of 1 nearest neighbour , this is because the training set is extremely small and hence we don't see much of a difference in the accuracies of them both.

We notice a trend where in higher the bin size , lower the accuracy i.e.

$bin\_size \propto 1/accuracy$



Trend Model