# Classification of images using K nearest neighbors

# Introduction

The classification problem is having 3 classes as night, portrait and landscape images. To get the prediction right for a random test image we need to create a dataset first. After the creation of this labelled dataset the test image is voted for y their K nearest neighbors, to see which category it belongs to.

## For this we used two approaches:

1. First approach was to compare the histograms.
2. Second was to compare the images directly by flattening the BGR matrix.

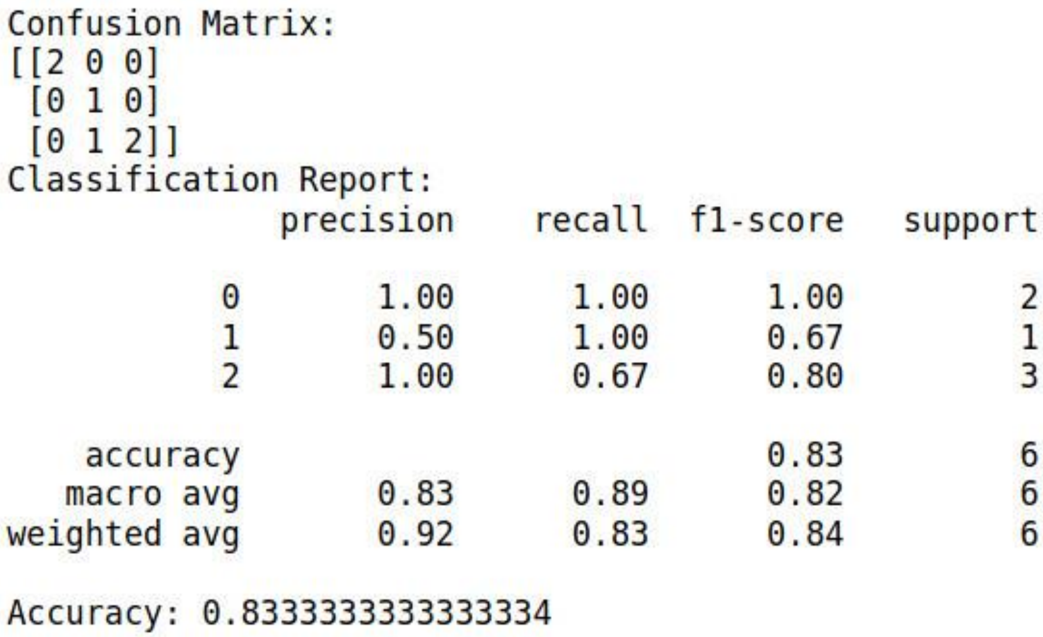
### Implimentation

The two approaches are compared simultaneously. The step followed are:

1. Creating a dataset with image matrix flattened and a histogram. The key was a string which represents the image name from the data set. And the images are classified into three classes with label encoding. Since the images were of different size, they were first converted into the same size of 256*256. Then the image matrix and histograms were calculated.
2. The dataset made is written into a csv file.
3. Then the multi class classifier KNN was trained using the histograms of the images.
4. Similarly, the classifier was trained for the flattened image.
5. After that the accuracy of both the classifier were compared.

## Results

The KNN classifier for both the types were compared by plotting the confusion matrix.



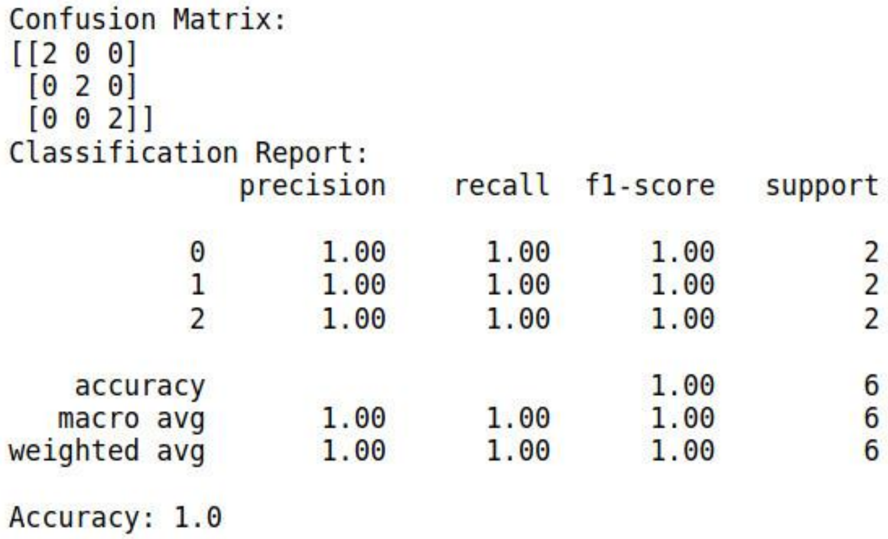
```
Confusion Matrix:
[[2 0 0]
 [0 1 0]
 [0 1 2]]
Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         2
           1       0.50      1.00      0.67         1
           2       1.00      0.67      0.80         3

    accuracy                           0.83         6
   macro avg       0.83      0.89      0.82         6
weighted avg       0.92      0.83      0.84         6

Accuracy: 0.8333333333333334
```

Fig a) Confusion matrix for KNN in histogram



```
Confusion Matrix:
[[2 0 0]
 [0 2 0]
 [0 0 2]]
Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         2
           1       1.00      1.00      1.00         2
           2       1.00      1.00      1.00         2

    accuracy                           1.00         6
   macro avg       1.00      1.00      1.00         6
weighted avg       1.00      1.00      1.00         6

Accuracy: 1.0
```

Fig b) Confusion matrix for flattened image matrix

Comparing the above two approaches the shows that, since the data set was small, image matrix KNN didn’t had the problem to guess the class to which the image belongs to. But in histogram approach it lost its spatial distribution and hence the accuracy went down.