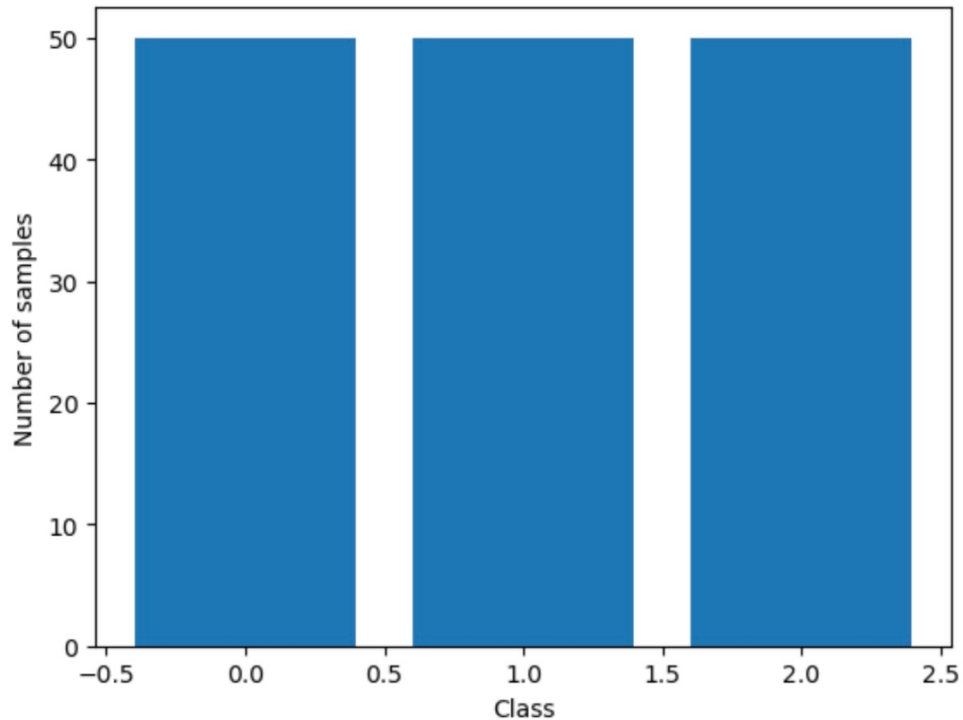


TODO Tasks for Submission

Aim: Use the 'Iris' dataset to perform KNN classification using 30% of the dataset as a test set.

Q1: Is your target class balanced? Explain your findings using a plot.



They are balanced, as each of the class has the same value which is 50.

Q2: Create a KNN classifier for the Iris dataset by using the best K (K ranges 1:30). Provide the plot of the train and test score for each k. Explain your findings.

```
test_scores = []
train_scores = []

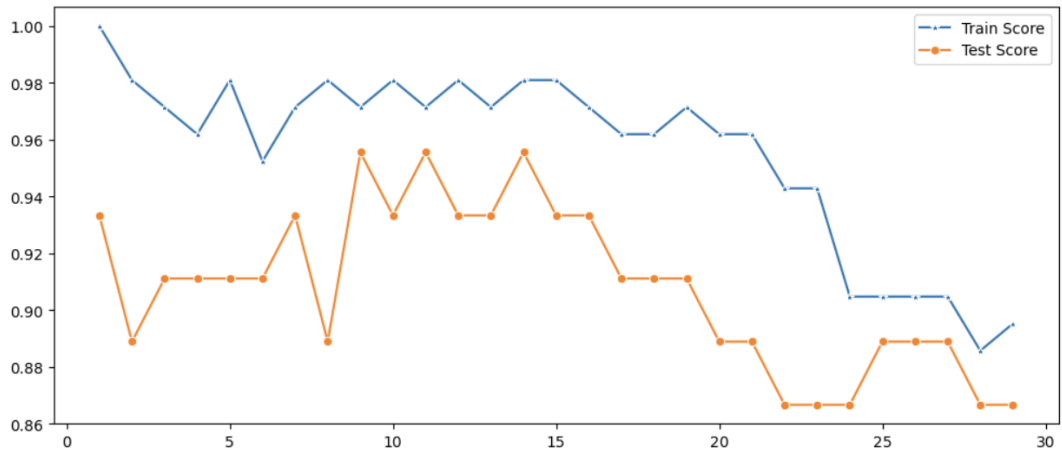
for i in range(1,30):

    knn = KNeighborsClassifier(i)
    knn.fit(X_train,y_train)

    train_scores.append(knn.score(X_train,y_train))
    test_scores.append(knn.score(X_test,y_test))

## Find the max score from testing on the same datapoints that were used for training
max_train_score = max(train_scores)
train_scores_ind = [i for i, v in enumerate(train_scores) if v == max_train_score]
print('Max train score {} % and k = {}'.format(max_train_score*100,list(map(lambda x: x+1, train_scores_ind))))
```

Max train score 99.16666666666667 % and k = [1]



Findings: The optimal value of k is the maximum point in the test score curve which is 9, 11, 14.

Q3: Create your final KNN (using the best k) and provide the decision regions plot.

Note: Do not submit your codes.

