Celestini Project 2019

Machine Learning Problem
(Section-B)

Problem Statement : To design a machine learning algorithm for finding the zoo for every animal , given the set of features of the animal.

Models to be used :

1. Multiclass SVM with the following kernels-
    - Linear Kernel
    - RBF Kernel
    - Polynomial Kernel
    - Sigmoid Kernel

2. Back Propagation Neural Network

Task 1:

Performing k-cross (k=5) validation on the dataset

Solution: Python code snippet and its output

```
# INPUT
scores = []
clf = OneVsRestClassifier(SVC(kernel='linear'))
# Using KFold  for cross validation with n_folds = 5
cv = KFold(n_splits=5, random_state=None, shuffle=False)
for train_index, test_index in cv.split(X):
    print("Train Index: ", train_index, "\n")
    print("Test Index: ", test_index)
    X_train, X_test, y_train, y_test = X[train_index], X[test_index], Y[train_index],
Y[test_index]
    clf.fit(X_train, y_train)
    scores.append(clf.score(X_test, y_test))
    y_score = clf.decision_function(X_test)
```

#OUTPUT

```
Train Index:  [ 21  22  23  24  25  26  27  28  29  30  31  32  33  34  35
36  37  38
  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56
  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74
  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92
  93  94  95  96  97  98  99 100]

Test Index:  [ 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19
20]
Train Index:  [  0   1   2   3   4   5   6   7   8   9  10  11  12  13  14
15  16  17
  18  19  20  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55
  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73
  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91
  92  93  94  95  96  97  98  99 100]

Test Index:  [21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40]
Train Index:  [  0   1   2   3   4   5   6   7   8   9  10  11  12  13  14
15  16  17
  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35
  36  37  38  39  40  61  62  63  64  65  66  67  68  69  70  71  72  73
  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91
  92  93  94  95  96  97  98  99 100]
Test Index:  [41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60]
Train Index:  [  0   1   2   3   4   5   6   7   8   9  10  11  12  13  14
15  16  17
  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35
  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53
  54  55  56  57  58  59  60  81  82  83  84  85  86  87  88  89  90  91
  92  93  94  95  96  97  98  99 100]
Test Index:  [61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80]
Train Index:  [ 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19
20 21 22 23
```

```
 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47
 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71
 72 73 74 75 76 77 78 79 80]

Test Index:  [ 81  82  83  84  85  86  87  88  89  90  91  92  93  94  95
96  97  98
  99 100]
```

Explanation : K-Fold is a cross validation technique in which a holdout set from the dataset is ignored and training is done on the rest part. This is done so that each part of data gets to be trained k-1 times.
For this problem ‘k’ is taken to be 5 which is why we get 5 different sets for training and test data. Accuracy and precision is computed over these and the average of each is then provided as evaluation metrics.