Pseudo Code –

1. Initialize all variables
2. Get top 6000 records of the training and the last 1000 records of the testing data in a List form
3. Define a function called Euclidean\_distance() which computes the Euclidean distance between two 1-D arrays. (i.e. the Euclidean distance between a training record and a test record).
4. Get square of difference between elements of the two lists.
5. Return the square root of sum of all the above values from (a).

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Testing\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. Loop for each test record (test\_i):
2. Call Euclidean\_distance() to calculate Euclidean distance between pixels of test\_i and all training records train\_i(Loop used for iterating each training record)
3. Sort distances from test\_i to each training record train\_i and stored in “one\_Test\_Point\_dist”
4. Loop for each K value of KNN classifier,
   1. Find K first lowest distances from the list “one\_Test\_Point\_dist”, i.e K nearest neighbors
   2. Get the mode of the labels of K nearest neighbors.
   3. Store the output of comparing the mode value and the class label of test\_i in List
5. Calculate accuracy for each K value of KNN classifier
6. Plot values of K v/s the testing error. (Testing Error = 1- accuracy)