Assignment no.B13
Roll no. 4234

1 Title

K-NN approach.

2 Problem Definition

Implementation of K-NN approach take suitable example.

3 Objective

- To understand the k-NN Approach
- To implement the k-NN using any suitable example

4 Software and Hardware Requirements

1. 64 bit Machine i3/i5/i7
2. 64-bit open source Linux OS Fedora 20
3. Python
4. Eclipse

5 Theory

The K Nearest Neighbor (k-NN) is a very intuitive method that classifies unlabeled examples based on their similarity with examples in the training set.k-NN is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until classification. The k-NN algorithm is among the simplest of all machine learning algorithms. In pattern recognition, the k-Nearest Neighbors algorithm (or k-NN for short) is a non-parametric method used for classification and regression. In both cases, the input consists of the k closest training examples in the feature space. The output depends on whether k-NN is used for classification or regression:

1. In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbor.

2. In k-NN regression, the output is the property value for the object. This value is the average of the values of its k nearest neighbors.

3. k-NN is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until classification. The k-NN algorithm is among the simplest of all machine learning algorithms.

- **k-NN advantages**

1. The cost of the learning process is zero

2. No assumptions about the characteristics of the concepts to learn have to be done

3. Complex concepts can be learned by local approximation using simple procedures

4. Robust to noisy training data

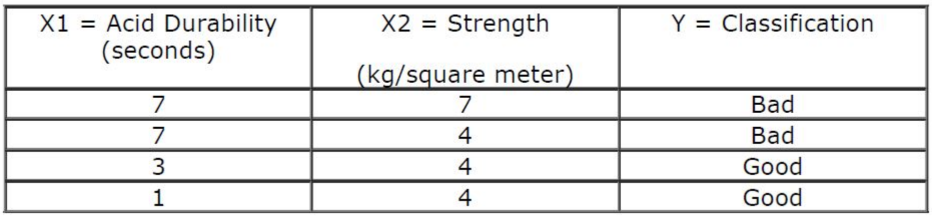
- **k-NN disadvantages**

1. Need to determine value of parameter k

2. Distance based learning is not clear which type of distance to use and which attribute to use to produce the best results.

3. Computation cost is quite high because we need to compute distance of each query instance to all the training samples.

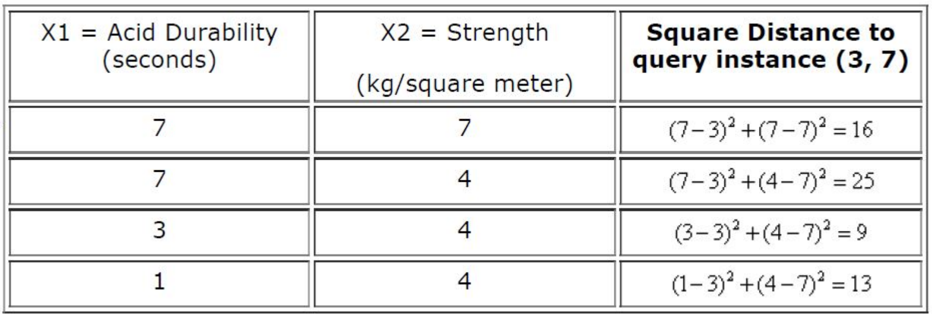
k-NN Example: 1 Let us suppose there is a data from the questionnaires survey (to ask people opinion) and objective testing with two attributes (acid durability and strength) to classify whether a special paper tissue is good or not. Here are four training samples Now the factory produces a new paper tissue that pass laboratory test with X1 = 3 and X2 = 7. Without another expensive survey, we need to guess what the classification of this new tissue is?



X1 = Acid Durability (seconds)	X2 = Strength (kg/square meter)	Y = Classification
7	7	Bad
7	4	Bad
3	4	Good
1	4	Good

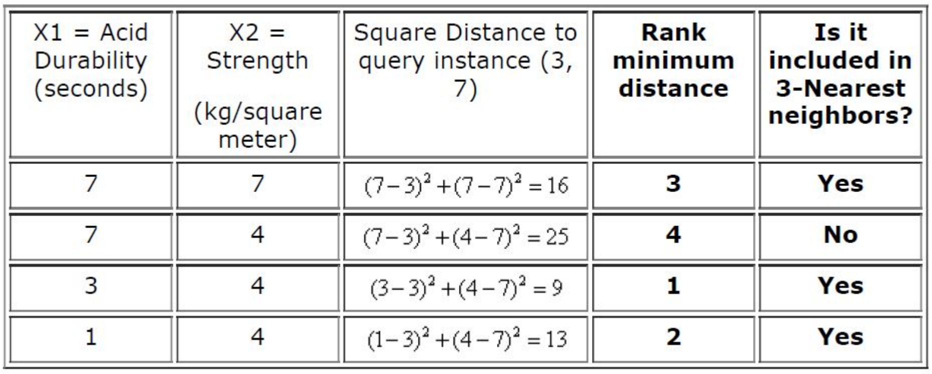
Solving the above problem using k-NN

1. Determine parameter K = number of nearest neighbors,Say K = 3
2. Calculate the distance between the query instance and all the training samples Coordinate of query instance is (3, 7), instead of calculating the distance compute square distance which is faster to calculate (without square root)



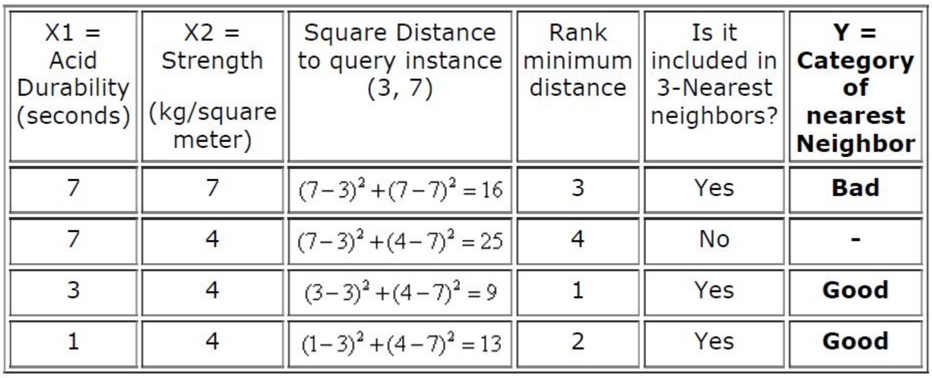
X1 = Acid Durability (seconds)	X2 = Strength (kg/square meter)	**Square Distance to query instance (3, 7)**
7	7	$(7-3)^2+(7-7)^2=16$
7	4	$(7-3)^2+(4-7)^2=25$
3	4	$(3-3)^2+(4-7)^2=9$
1	4	$(1-3)^2+(4-7)^2=13$

3. Sort the distance and determine nearest neighbors based on the K-th minimum distance



X1 = Acid Durability (seconds)	X2 = Strength (kg/square meter)	Square Distance to query instance (3, 7)	**Rank minimum distance**	**Is it included in 3-Nearest neighbors?**
7	7	$(7-3)^2+(7-7)^2=16$	**3**	**Yes**
7	4	$(7-3)^2+(4-7)^2=25$	**4**	**No**
3	4	$(3-3)^2+(4-7)^2=9$	**1**	**Yes**
1	4	$(1-3)^2+(4-7)^2=13$	**2**	**Yes**

4. Gather the category Y of the nearest neighbors. Notice in the second row last column that the category of nearest neighbor (Y) is not included because the rank of this data is more than 3 (=K).



X1 = Acid Durability (seconds)	X2 = Strength (kg/square meter)	Square Distance to query instance (3, 7)	Rank minimum distance	Is it included in 3-Nearest neighbors?	**Y = Category of nearest Neighbor**
7	7	$(7-3)^2+(7-7)^2=16$	3	Yes	**Bad**
7	4	$(7-3)^2+(4-7)^2=25$	4	No	**-**
3	4	$(3-3)^2+(4-7)^2=9$	1	Yes	**Good**
1	4	$(1-3)^2+(4-7)^2=13$	2	Yes	**Good**

5. Using simple majority of the category of nearest neighbors as the prediction value of the query instance.
   Since We have 2 good and 1 bad, since 2 is greater than 1 then we conclude that a new paper tissue that pass laboratory test with X1 = 3 and X2 = 7 is included in **Good** category.

6 Mathematical Model

Let S be the solution perspective of the class k-NN such that
S={s, e, i, o, f, DD, NDD, success, failure}

For class k-NN
s={Initial state of the class}

e={End state or destructor of the class}

i={I1}
I1={x—x is input text provided to the system present in the document to parse.}

o={o1}
o1={x—x is the preprocessed data and calculated euclidean distance with respect to document.}

F_{me}=set of functions.

F_{me}={f1,f2,f3,f4}
where,
f1= f1 represents the function to read the input from a file .

f2= f2 represents the function to calculate the euclidean distance .

f3= f3 represents the function to get neighbouring nodes.

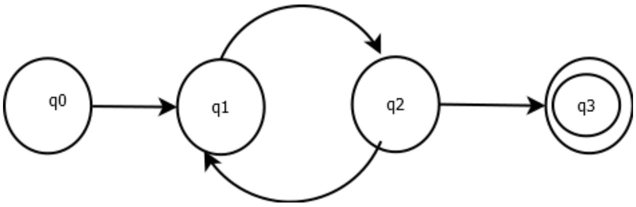
f4 = f4 to prints the majority of the category of nearest neighbours as the prediction value of the query instance

DD (Deterministic Data) = input file

NDD (Non Deterministic Data) = the distance calulated

Sc = category of the nearest neighbour.

7 State Diagram :



q0 → start state. Accept the file as a input

q1 → calculate the distance euclidean distance

q2 → to get neighbouring nodes

q3 → to prints the majority of the category of nearest neighbours.

8 Algorithm

1. start
2. Determine parameter K = number of nearest neighbors
3. Calculate the distance between the query instance and all the training samples
4. Sort the distance and determine nearest neighbors based on the K-th minimum distance
5. Gather the category Y of the nearest neighbors
6. Use simple majority of the category of nearest neighbors as the prediction value of the query instance
7. stop

9 Conclusion

Thus we studied k-NN approach for classification and implemented it by taking a suitable example and predicted results based on it.