Pseudo Code

MAIN( )

1. X , Y = Preprocessing ( )
2. Input (I , k , T)
3. Initialize missing data with avg features and $C = 11^T$
4. loop1 for t = 1:T do
5. loop2 till convergence of X^{ny} and Y^{nx} do
6. $P = diag(C)HXX^TH$
7. first fix all and update Y^{nx} using eqn $Y^{nx} = -(P^{nxnx})^{-1} (P^{ncnx})^T Y^{nc} + P^{nxny} Y^{ny})$
8. $Q = HYY^Tdiag(C)H$
9. first fix all and update X^{nx} using eqn $X^{ny} = -(P^{nyny})^{-1} (P^{ncny})^T X^{nc} + P^{nxny} X^{nx})$
10. loop2 end
11. list1 = KNN(X)
12. loop for i = 1:n do
13. loop for j = 1:n do
14. if (i belongs to nearest neighbors of j) then
15. $W^x[i,j] = 1$
16. else
17. $W^x[i,j] = 0$
18. end for
19. end for
20. list2 = KNN(Y)
21. loop for i = 1:n do
22. loop for j = 1:n do
23. if (i belongs to nearest neighbors in list[j]) then
24. $W^y[i,j] = 1$
25. else
26. $W^y[i,j] = 0$
27. end for
28. end for
29. $delta = HW^xHW^y$
30. loop for i = 1:n do
31. S[i] = delta [i,i]
32. end for

33. scale S btw {0.1 - 1}
34. $C = SS^T$
35. loop for i = 1:n do
36. if (S[i] < T)
37. O[i] = 1
38. else
39. O[i] = 0
40. end for

KNN(X,Y)

1. loop x_i i = 1:len(n)
2. Calculate "d(x, x_i)" i =1, 2, ….., **n**; where **d** denotes the Euclidean distance between the points.
3. Arrange the calculated **n** Euclidean distances in non-decreasing order.
4. Let **k** be a +ve integer, take the first **k** distances from this sorted list.
5. Find those **k**-points corresponding to these **k**-distances.
6. Let $\mathbf{k}_i$ denotes the number of points belonging to the i^{th} class among **k** points i.e. $k \geq 0$
7. If $k_i > k_j \sum i \neq j$ then put x in list(i).

nc = 1:300
nx = 301:400
ny = 401:500

I = self guided iterations

k = nearest neighbors

T = threshold

eqn $= Y^{nx} = -(P^{nxnx})^{-1} (P^{ncnx})^T Y^{nc} + P^{nxny} Y^{ny}$ obtained by setting derivative w.r.t Y^{nx} to zero of max Y^{nx} { $Tr [Y^{nx} (Y^{nx})^T P^{nx\,nx}] + 2*Tr [Y^{nx} (Y^{nc})^T P^{nc\,nx}] + 2*Tr [Y^{ny} (Y^{nx})^T P^{nx\,ny}]$ })