```
S = load('wikipedia_m.mat');
disp(S);
```

```
      articles: {500×1 cell}
    dictionary: {4423×1 cell}
      tdmatrix: [4423×500 double]
```

```

if isfield(S, 'tdmatrix')
    X = S.tdmatrix;
    articles = S.articles;
    dictionary = S.dictionary;
else
    X = tdmatrix;
end

[d,N] = size(X);

k = 8;
i = 200;
error_tol = 1e-8;
numR = 5;

bestJ = inf;
best = struct();

for r = 1:numR
    fprintf("Restart %d/%d", r, numR);
    group = randi(k, 1, N);
    z = zeros(d, k);
    Jprev = inf;

    for j = 1:i
        for g = 1:k
            idx = find(group == g);
            if ~isempty(idx)
                z(:,g) = mean(X(:,idx),2);
            else
                z(:,g) = X(:,randi(N));
            end
        end

        dists = zeros(k, N);
        for g = 1:k
            D = X - z(:,g);
            dists(g,:) = sum(D.^2, 1);
        end
        [mind2, newGroup] = min(dists, [], 1);
        J = mean(mind2);

        if j > 1 && abs(J - Jprev) <= error_tol * Jprev
```

```
            fprintf("Converged in %d i. J = %.6g\n", j, J);
            break;
        end

        group = newGroup;
        Jprev = J;
        if j == i
            fprintf("Reached i. J = %.6g\n", J);
        end
    end

    if J < bestJ
        bestJ = J;
        best.z = z;
        best.group = group;
        best.J = J;
        best.dists = dists;
    end
end
```

```
Restart 1/5
Converged in 16 i. J = 0.00708804
Restart 2/5
Converged in 9 i. J = 0.00709056
Restart 3/5
Converged in 15 i. J = 0.00699051
Restart 4/5
Converged in 19 i. J = 0.00711807
Restart 5/5
Converged in 9 i. J = 0.00709408
```

```
fprintf("Best objective across restarts: J = %.6g", bestJ);
```

```
Best objective across restarts: J = 0.00699051
```

```
z = best.z;
group = best.group;

topTerms = cell(k,1);
closestArticle = cell(k,1);

for g = 1:k
    [~, ord] = sort(z(:,g), 'descend');
    topIdx = ord(1:min(5, numel(ord)));
    topTerms{g} = dictionary(topIdx);

    idx = find(group == g);
    if isempty(idx)
        closestArticle{g} = {'(empty cluster)'};
    else
        Dg = X(:,idx) - z(:,g);
        dd = sum(Dg.^2, 1);
        [~, loc] = sort(dd, 'ascend');
        pick = idx(loc(1:min(5, numel(loc))));
        closestArticle{g} = articles(pick);
    end
end

for g = 1:k
    fprintf(' Cluster %d\n', g);
    fprintf(' Top Terms: ');
    fprintf(' %s ', topTerms{g}{:});
    fprintf('\n Closest articles:\n');
    for j = 1:numel(closestArticle{g})
        fprintf('- %s\n', closestArticle{g}{j});
    end
end
```

```
 Cluster 1
 Top Terms:
 series  season  episode  film  television
 Closest articles:
- The_X-Files
- Charlie_Sheen
- Game_of_Thrones
- House_of_Cards_(U.S._TV_series)
- Supergirl_(U.S._TV_series)
 Cluster 2
 Top Terms:
 album  release  song  music  single
 Closest articles:
- David_Bowie
- Kanye_West
- Celine_Dion
- Ariana_Grande
- Kesha
 Cluster 3
 Top Terms:
 film  star  million  role  release
 Closest articles:
- Leonardo_DiCaprio
- Kate_Beckinsale
- Star_Wars:_The_Force_Awakens
- Star_Wars_Episode_I:_The_Phantom_Menace
- Maureen_O'Hara
 Cluster 4
 Top Terms:
 game  match  team  player  play
 Closest articles:
- Halo_5:_Guardians
- Fallout_4
- Call_of_Duty:_Black_Ops_III
- Overwatch_(video_game)
- Call_of_Duty:_Modern_Warfare_2
 Cluster 5
 Top Terms:
 match  championship  event  style  raw
 Closest articles:
- Wrestlemania_32
- Payback_(2016)
- Royal_Rumble_(2016)
- Night_of_Champions_(2015)
- Survivor_Series_(2015)
 Cluster 6
 Top Terms:
 season  win  game  team  player
 Closest articles:
- Kobe_Bryant
- Lamar_Odom
- Jose_Mourinho
- Johan_Cruyff
- Tom_Brady
 Cluster 7
 Top Terms:
 united  family  american  city  national
 Closest articles:
- Mahatma_Gandhi
- Sigmund_Freud
- Ben_Affleck
- Carly_Fiorina
- Frederick_Douglass
 Cluster 8
 Top Terms:
 fight  win  event  champion  fighter
 Closest articles:
- Floyd_Mayweather,_Jr.
- Kimbo_Slice
- Ronda_Rousey
- Jose_Aldo
- Joe_Frazier
```