
Table of Contents

.....	1
Spectral clustering	1
(a) Purity metric	2
(b) Plot clusters for $K = 5$	3

```
% Austin Welch
% EC503 HW8.2
% Spectral Clustering on Airbnb Data
% http://insideairbnb.com/get-the-data.html

clear all; clc; %#ok<*CLALL>
Data = load('BostonListing.mat');
X = [Data.latitude, Data.longitude];
Y = Data.neighbourhood;
```

Spectral clustering

Gaussian similarity distance, fully-connected graph, std = 0.01, symmetrically normalized graph Laplacian for spectral clustering for $K=1,2,\dots,25$ calculate the "purity" metric of the obtained cluster by treating the neighborhood label as the ground truth. Plot the purity metric (y-axis) as a function of k (x-axis)

```
% Step 1: Weighted adjacency matrices

% fully-connected graph, W = S (similarity score)
W = zeros(length(X),length(X));
sigma = 0.01; % given std

for i=1:length(X)
    for j=1:length(X)
        % calculate gaussian similarity scores S(xi,xj)
        W(i,j) = exp(-((X(i,1)-X(j,1))^2 + (X(i,2)-X(j,2))^2)/ ...
            (2*sigma.^2));
    end
end

% Step 2: Degree matrix D
D = diag(sum(W,2));

% Compute the un-normalized graph Laplacian L = D - W
L = D - W;

% Compute the normalized graph Laplacian L_sym = D^{-1/2} * L * D^{-1/2}
L_sym = inv(sqrt(D))*L*inv(sqrt(D)); %#ok<MINV>

[V,G] = svd(L_sym);
```

```
K = 1:25;
Vmats = cell(length(K),1);

for i=K
    Vmats{i} = V(:,end-i+1:end);
end

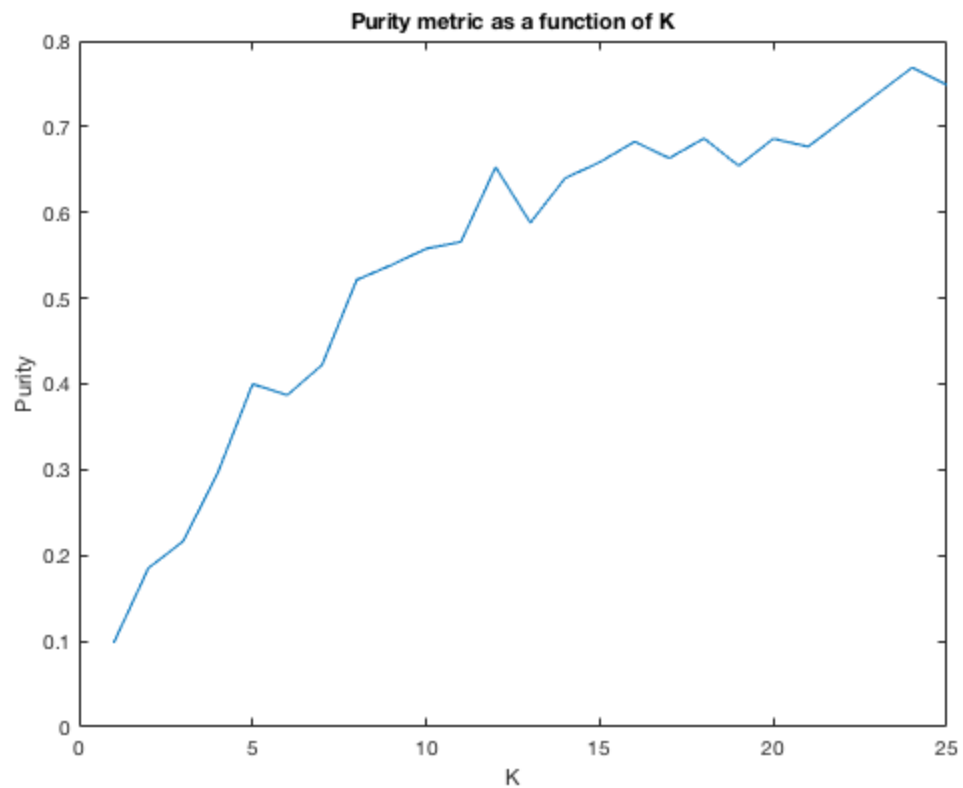
% normalize V rows so that l-2 norms are 1
for i=K % implicit expansion: Matlab 2016b+
    Vmats{i} = Vmats{i} ./ sqrt(sum(Vmats{i}.^2,2));
end

% perform K-means on V mats
P = cell(25,1);
for i=K
    rng(2);
    P{i} = kmeans(Vmats{i},i);
end
```

(a) Purity metric

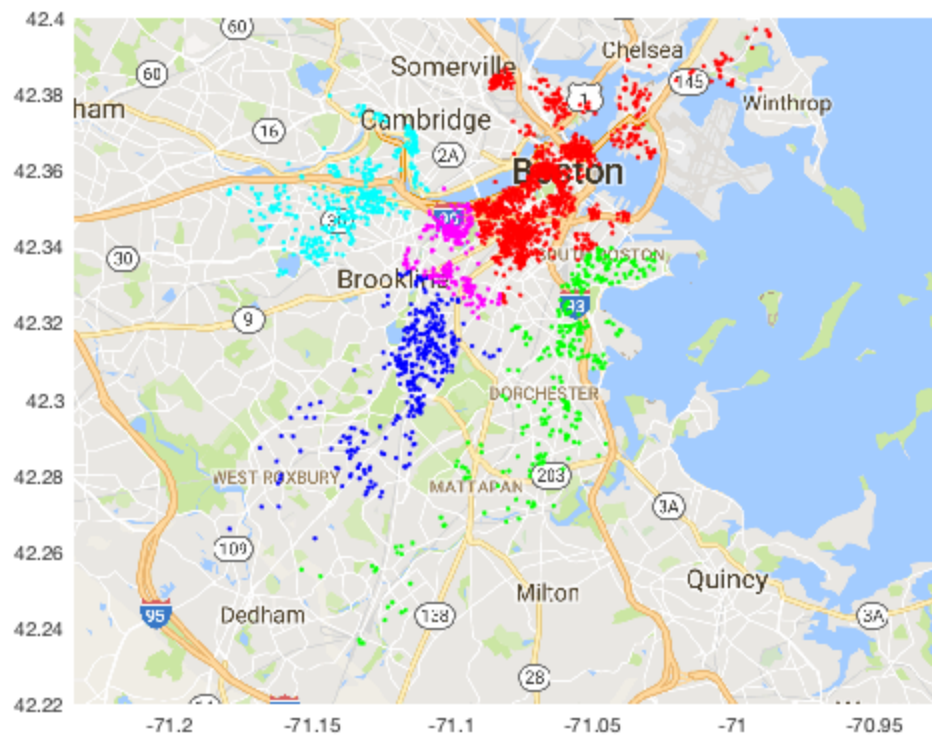
```
purity = zeros(25,1);
for i=K
    votes = cell(i,1);
    num = 0;
    denom = 0;
    for j=1:i
        Ysub = Y(P{i}==j);
        [unique_strings,~,string_map] = unique(Ysub);
        most_common_string = unique_strings(mode(string_map));
        votes{j} = most_common_string;

        num = num + sum(strcmp(most_common_string,Ysub));
        denom = denom + length(Ysub);
    end
    purity(i) = num/denom;
end
figure(1);
plot(K,purity);
title('Purity metric as a function of K');
xlabel('K');
ylabel('Purity');
```



(b) Plot clusters for K = 5

```
figure(2); hold on;  
lat = X(:,1); lon = X(:,2);  
plot(lon(P{5}==1),lat(P{5}==1),'.r','MarkerSize',6)  
plot(lon(P{5}==2),lat(P{5}==2),'.b','MarkerSize',6)  
plot(lon(P{5}==3),lat(P{5}==3),'.g','MarkerSize',6)  
plot(lon(P{5}==4),lat(P{5}==4),'.m','MarkerSize',6)  
plot(lon(P{5}==5),lat(P{5}==5),'.c','MarkerSize',6)  
plot_google_map
```



Published with MATLAB® R2017a