ISYE 6740 Homework 1 solution

1 Clustering [25 points]

Given m data points x^i, $i = 1, \ldots, m$, K-means clustering algorithm groups them into k clusters by minimizing the distortion function over $\{r^{ij}, \mu^j\}$

$$J = \sum_{i=1}^{m} \sum_{j=1}^{k} r^{ij} \|\mathrm{x}^i - \mu^j\|^2,$$

where $r^{ij} = 1$ if x^i belongs to the j-th cluster and $r^{ij} = 0$ otherwise.

1. (10 points) Prove (using mathematical arguments) that using the squared Euclidean distance $\|\mathrm{x}^i - \mu^j\|^2$ as the dissimilarity function and minimizing the distortion function, we will have

$$\mu^j = \frac{\sum_i r^{ij} \mathrm{x}^i}{\sum_i r^{ij}}.$$

That is, μ^j is the center of j-th cluster.

$$\begin{aligned} \frac{\partial J}{\partial \mu^j} &= \sum_{i=1}^{m} 2 r^{ij} (x^i - \mu^j) \\ &= 2 \sum_i r^{ij} x^i - 2 \sum_i r^{ij} \mu^j = 0 \\ \Rightarrow \quad \sum_i r^{ij} x^i &= \sum_i r^{ij} \mu^j \\ \mu^j &= \frac{\sum_i r^{ij} \mathrm{x}^i}{\sum_i r^{ij}} \end{aligned}$$

since the above squared Euclidean distance is a quadratic function of μ^j, we hence conclude the statement.

rubric: any reasonable attempt 3pts, correct proof 10pts

2. (5 points) Prove (using mathematical arguments) that K-means algorithm converges to a local optimum in finite steps.

proof sketch:
1.There are limit number of total possible combination of the cluster assignments to the certain number of data points.
2. During each iteration, the cost function decreases monotonically.

rubric: any reasonable attempt 1pts, address each of the above points 2pts

3. (10 points) Calculate k-means by hands. Given 5 data points configuration in Figure 1. Assume $k = 2$ and use Manhattan distance (a.k.a. the ℓ_1 distance: given two 2-dimensional points (x_1, y_1) and (x_2, y_2), their distance is $|x_1 - x_2| + |y_1 - y_2|$). Assuming the initialization of centroid as shown, after one iteration of k-means algorithm, answer the following questions.

   (a) Show the cluster assignment;

   (b) Show the location of the new center;

   (c) Will it terminate in one step?

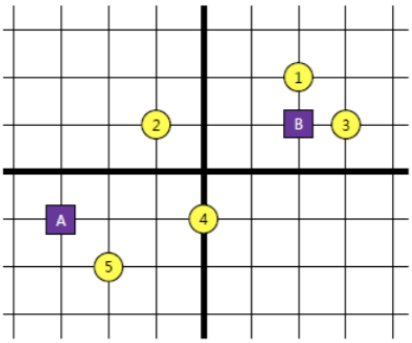


Figure 1: K-means.

(a) the new cluster assignment: $A = \{4, 5\}$, $B = \{1, 2, 3\}$

(b) the new centers: To solve the new centers, we follow the definition, and solve the optimization problem. For the first updated centroid:

$$C_A = \arg \min_{v_1 \in \mathbb{R}, v_2 \in \mathbb{R}} |v_1 - 0| + |v_2 + 1| + |v_1 + 2| + |v_2 + 2|$$

Note that the objective function is decoupled in v_1 and v_2, so we can solve two one-dimensional optimization problem with respect to each of them separately; the plot is shown in Figure 2. Note that the optimization is convex; in this case, they are simple and we can derive the solution. Note that any $v_1 \in [-2, 0]$ and $v_2 \in [-2, -1]$ will minimize the function. So you can pick one minimizer as the solution, for example: $C_A = (-1, -\frac{3}{2})$.

For the second update centriod:

$$C_B = \arg \min_{v_1 \in \mathbb{R}, v_2 \in \mathbb{R}} |v_1 - 2| + |v_2 + 2| + |v_1 + 1| + |v_2 - 1| + |v_1 - 3| + |v_2 - 1|$$

From the plot in Figure 3, we find that the minimizer is $v_1 = 2$ and $v_2 = 1$. Note that this anticipated, since the optimization problem here is a linear objective function, and the minimizer should happens at one of the vertices. As the result, the update centroid is: $C_B = (2, 1)$.

(c) the new cluster assignment after one iteration:
cluster A: {2,4,5} cluster B: {1,3}
hence algorithm will not terminate

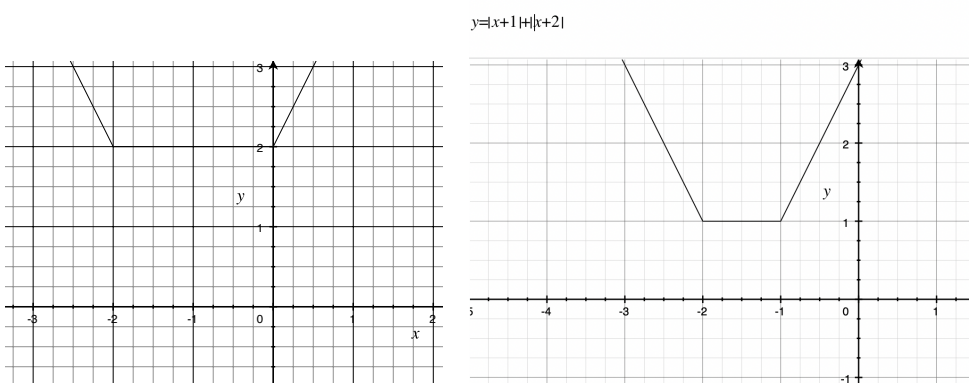


Figure 2: Left: Plot of $y = |x| + |x + 2|$; note that any $x \in [-2, 0]$ will minimize the function; Right: Plot of $y = |x + 1| + |x + 2|$; note that any $x \in [-2, -1]$ will minimize the function.

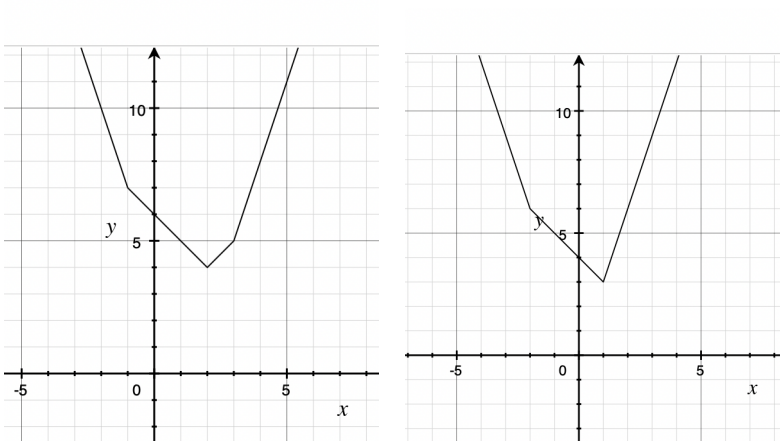


Figure 3: Left: Plot of $y = |x - 2| + |x + 1| + |x - 3|$ and the minimizer is $x = 2$; Right: Plot of $y = |x + 2| + |x - 1| + |x - 1|$ and the minimizer is $x = 1$.

rubric:

(a) any attempt 1pt, correct answer 4pts

(b) any attempt 1pt, correct answer 4pts

(c) answer 'No' 2pts. No detailed procedure/explanation required

2 Image compression using clustering [25 points]

In this programming assignment, you are going to apply clustering algorithms for image compression. Your task is implementing the clustering parts with two algorithms: *K-means* and *K-medoids*. **It is required you implementing the algorithms yourself rather than calling from a package.**

K-medoids

In class, we learned that the basic K-means works in Euclidean space for computing distance between data points as well as for updating centroids by arithmetic mean. Sometimes, however, the dataset may work better with other distance measures. It is sometimes even impossible to compute arithmetic mean if a feature is categorical, e.g, gender or nationality of a person. With K-medoids, you choose a representative data point for each cluster instead of computing their average. Please note that K-medoid is different from generalized K-means: Generalized K-means still computes centre of a cluster is not necessarily one of the input data points (it is a point that minimizes the overall distance to all points in a cluster in a chosen distance metric).

Given m data points $\mathrm{x}^i (i = 1, \ldots, m)$, K-medoids clustering algorithm groups them into K clusters by minimizing the distortion function $J = \sum_{i=1}^{m} \sum_{j=1}^{k} r^{ij} D(\mathrm{x}^i, \mu^j)$, where $D(\mathrm{x}, \mathrm{y})$ is a distance measure between two vectors x and y in same size (in case of K-means, $D(x, y) = \|\mathrm{x} - \mathrm{y}\|^2$), μ^j is the center of j-th cluster; and $r^{ij} = 1$ if x^n belongs to the k-th cluster and $r^{ij} = 0$ otherwise. In this exercise, we will use the following iterative procedure:

- Initialize the cluster center μ^j, $j = 1, ..., k$.
- Iterate until convergence:
  - Update the cluster assignments for every data point x^i: $r^{ij} = 1$ if $j = \arg\min_j D(\mathrm{x}^i, \mu^j)$, and $r^{ij} = 0$ otherwise.
  - Update the center for each cluster j: choosing another representative if necessary.

There can be many options to implement the procedure; for example, you can try many distance measures in addition to Euclidean distance, and also you can be creative for deciding a better representative of each cluster. We will not restrict these choices in this assignment. You are encouraged to try many distance measures as well as way of choosing representatives (e.g., ℓ_1 norm).

Formatting instruction

Input

- `pixels`: the input image representation. Each row contains one data point (pixel). For image dataset, it contains 3 columns, each column corresponding to Red, Green, and Blue component. Each component has an integer value between 0 and 255.
- `k`: the number of desired clusters. Too high value of K may result in empty cluster error. Then, you need to reduce it.

Output

- `class`: cluster assignment of each data point in pixels. The assignment should be 1, 2, 3, etc. For $k = 5$, for example, each cell of class should be either 1, 2, 3, 4, or 5. The output should be a column vector with `size(pixels, 1)` elements.
- `centroid`: location of k centroids (or representatives) in your result. With images, each centroid corresponds to the representative color of each cluster. The output should be a matrix with K rows and 3 columns. The range of values should be [0, 255], possibly floating point numbers.

Hand-in

Both of your code and report will be evaluated. Upload them together as a zip file. In your report, answer to the following questions:

1. (5 points) Within the k-medoids framework, you have several choices for detailed implementation. Explain how you designed and implemented details of your K-medoids algorithm, including (but not limited to) how you chose representatives of each cluster, what distance measures you tried and chose one, or when you stopped iteration.

   The general algorithm procedure:

   - Initialize the cluster center μ^j, $j = 1, \cdots, k$
   - iterate until convergence
     - Update the cluster assignments for every data point x^i: $r^{ij} = 1$ if $j = \arg\min_j D(x^i, \mu^j)$, and $r^{ij} = 0$ otherwise.
     - Update the center for each cluster j: choosing another representative if necessary

   Implementation details:

   - the representatives, i.e., the centroids, of a cluster, should be chosen as one of the data point that minimize the sum of the distance within the cluster. (This is the main difference from Kmeans algorithm, whose cluster centroid is the mean of the data point in the cluster, it would be different from any of the data point.)
   - the distance measure can be chosen from any reasonable distance metric, such as L_p norm (for continuous data), hamming distance (for categorical data), and etc
   - the convergence, i.e., iteration terminating criteria can be set as the cost no longer decrease

   $$R(X) = \sum_{i=1}^{m} d(x_i, C_{x_i})$$

   where $D(x_i), C_{x_i}$ is the distance metric function, C_{x_i} is the centroid associated with x_i

   rubric: any reasonable attempt 2pt. each answer to above three question 1pt

2. (10 points) Attach a picture of your own. We recommend size of 320×240 or smaller. Run your k-medoids implementation with the picture you chose, as well as two pictures provided (beach.bmp and football.bmp), with several different K. (e.g, small values like 2 or 3, large values like 16 or 32) What did you observe with different K? How long does it take to converge for each K? Please write in your report.

   reference result as below. the actual number of iteration/running time varies among different implementation efficiency, hardware, and testing image.

	Kmedoids		Kmeans	
K	iterations	times	iterations	times
2	4	0.62s	21	2.78s
4	3	0.47s	15	2.18s
8	5	0.89s	29	4.86s
16	11	2.70s	91	19.51s

   rubric:
   any program code that produce result without error, 2pts.
   reasonable result for: at least three value of K, for at least one provided image and student's own image, 6pts.
   proper analysis to the result including the running time 2pts

3. (5 points) Run your k-medoids implementation with different initial centroids/representatives. Does it affect final result? Do you see same or different result for each trial with different initial assignments? (We usually randomize initial location of centroids in general. To answer this question, an intentional poor assignment may be useful.) Please write in your report.

   In principle, k-medoids algorithm converges only at local optimum. Different initialization may end up with different result. Different initialization could lead to different running time. Some extreme initialization such as $[255, 255, 255]$ may cause the program to converge much slower. However, the output image would not have too much perceptible difference among the different initializations.

   rubric:
   reasonable effort 5pts

4. (5 points) Repeat question 2 and 3 with k-means. Do you see significant difference between K-medoids and k-means, in terms of output quality, robustness, or running time? Please write in your report.

Please refer to the next page for the output images

	Kmedoids		Kmeans	
K	iterations	times	iterations	times
2	4	0.62s	21	2.78s
4	3	0.47s	15	2.18s
8	5	0.89s	29	4.86s
16	11	2.70s	91	19.51s

rubric:
any program code that produce result without error 2pts.
reasonable result for: at least three value of K, for at least one provided image and student's own image, 2pts.
proper analysis 1pt

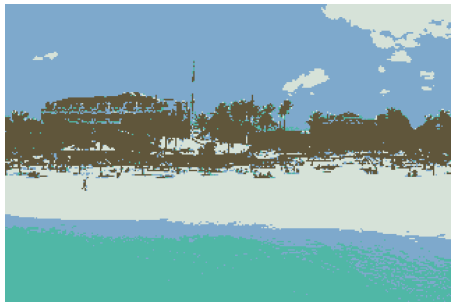
Note

- You may see some error message about empty clusters when you use too large k. Your implementation should treat this exception as well. That is, do not terminate even if you have an empty cluster, but use smaller number of clusters in that case.
- We will grade using test pictures which are not provided. We recommend you to test your code with several different pictures so that you can detect some problems that might happen occasionally.
- If we detect copy from any other student's code or from the web, you will not be eligible for any credit for the entire homework, not just for the programming part. Also, directly calling built-in functions or from other package functions is not allowed.

K-medoids K=2

K-means K=2

K-medoids K=4



K-means K=4

K-medoids K=8

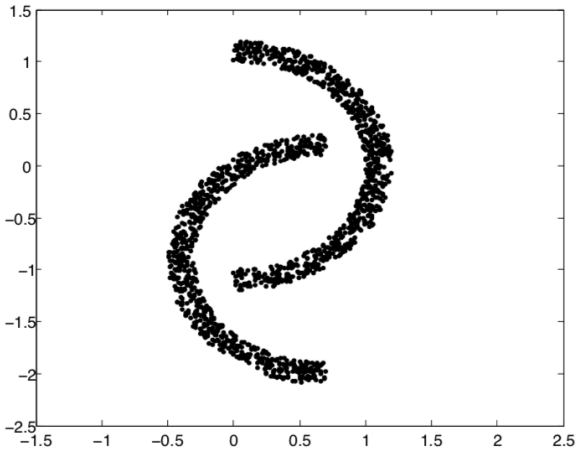
K-means K=8

K-medoids K=16

K-means K=16

3 Spectral clustering [25 points]

1. (10 points) For the following data (two moons), give one method that will successfully separate the two moons? Explain your rationale.



Spectral clustering algorithm can separate the two moons.
In the first step of spectral clustering algorithm, i.e., we capture the local connectivity relationship by building the adjacency matrix.

Note that an important property of graph Laplacian states that, *'the number of connected components in the graph is the dimension of the nullspace of the Laplacian and the algebraic multiplicity of the 0 eigenvalue'*. This property can be understood as that, the nullspace of the Laplacian captures the group(cluster) information of the vertex from the adjacency matrix.

Therefore, at the later part of the spectral clustering algoirthm, we can project the adjacency matrix on the nullspace (which corresponds to those eigenvector with eigenvalue 0), and perform kmeans algorithm on those projected vectors to find the cluster.

rubric:
correct choice of algorithm 5pts.
reasonable explanation 5pts.

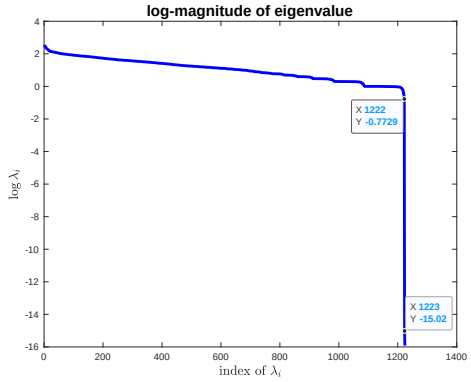
2. (15 points) Political blogs dataset.

   We will study a political blogs dataset first compiled for the paper Lada A. Adamic and Natalie Glance, "The political blogosphere and the 2004 US Election", in Proceedings of the WWW-2005 Workshop on the Weblogging Ecosystem (2005). The dataset nodes.txt contains a graph with $n = 1490$ vertices ("nodes") corresponding to political blogs. Each vertex has a 0-1 label (in the 3rd column) corresponding to the political orientation of that blog. We will consider this as the true label and try to reconstruct the true label from the graph using the spectral clustering on the graph. The dataset edges.txt contains edges between the vertices. You may remove isolated nodes (nodes that are not connected any other nodes).

   (a) (10 points) Assume the number of clusters to be estimated is $k = 2$. Using spectral clustering to find the 2 clusters. Compare the clustering results with the true labels. What is the false

classification rate (the percentage of nodes that are classified incorrectly). **It is required you implementing the algorithms yourself rather than calling from a package.**

We can follow the spectral clustering procedure as explained on the lecture slides.
The first step is build the adjacency matrix (A) based on the edges information from 'edges.txt'. Note that there are three nodes are isolated, they are node #24, #1047, and # 1260. After removing those isolated nodes, (the corresponding columns and rows from the A matrix) we have the updated A matrix of size 1487×1487. We can then calculate the Laplacian matrix L accordingly.



With SVD, we have the eigenvalues of the Lapacian L as the above image. Please note that the magnitude of the eigenvalues have been taken log of. We can see that the nullspace of Laplacian has size of only 2 columns.
Next, we can use a certain number of columns from the nullspace, perform *Kmeasns* and compare the labels with the true labels from the second column of 'nodes.txt' The accuracy result is 52.14%

rubric:
any reasonable efforts (if the code has error and does not produce result) 2pts
partial result: adjacency matrix A degree matrix D, graph Laplacian matrix L, 2pts each.
Kmeans results: accuracy rate above 50% 2pts

(b) (5 points) You might observe the performance is not as good as you expected (given that there is no coding bugs). What do you think might be the reason for the not-so-good performance, due to the discrepancy from "theory" and "application"? Please write in your report.

- We can see the nullspace of the graph Laplacian has only two columns. This implies it is capable to find the group configuration.
- The final classification result is very poor due to the model mismatch.
  By using the spectral clustering algorithm to investigate the political orientation of the blogs, we are assuming that, their community status, i.e., the graph clustering conditions, has strong enough correlation to their political orientation. This may not be the case.

rubric:
any reasonable explanation that demonstrates proper understanding to the algorithm / data 5pts

4 PCA: Food consumption in European area [25 points]

The data food-consumption.csv contains 16 countries in the European area and their consumption for 20 food items, such as tea, jam, coffee, yoghurt, and others. There are some missing data entries: you may remove the rows "Sweden", "Finland", and "Spain". The goal is to perform PCA analysis on the data, i.e., find a way to perform linear combinations of features across all 20 food-item consumptions, for each country. If we extract two principal components, that means we use two singular vectors that correspond to the largest singular values of the data matrix, in combining features.

1. (5 points) Write down the set-up of PCA for this setting. Explain how the data matrix is set-up in this case (e.g., each dimension of the matrix correspond to what.) Explain in words how PCA is performed in this setting.
   To proceed with PCA, we first define the data $X = \{x_i \in \mathbb{R}^d, i = 1 \cdots, m\}$, where the total number of data points $m = 13$, data dimension $d = 20$.
   The PCA procedure:

   - find the covariance matrix

   $$C = \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu)(x_i - \mu)^T$$
   $$\text{where} \qquad \mu = \frac{1}{m}\sum_{i=1}^{m} x_i$$

   - find the eigenvectors

   $$w_i: \quad \lambda_i C = C w_i, \quad i = 1, \cdots d$$

   where λ_i are corresponding eigenvalue of C

   - project the data onto the eigenvectors and scale by corresponding eigvenvalue

   $$z_i = \begin{bmatrix} w_1^T(x_i - \mu)/\sqrt{\lambda_1} \\ w_2^T(x_i - \mu)/\sqrt{\lambda_2} \\ \vdots \\ w_k^T(x_i - \mu)/\sqrt{\lambda_k} \end{bmatrix}$$

   where $k \leq d$ is the truncation number, i.e., number of eigenvectors chosen for the projection.

   rubric:
   proper data setting 2pts; correct procedure 3pts.

2. (5 points) Suppose we aim to find top k principal components. Write down the mathematical optimization problem involved for solving this problem. Explain the procedure to find the top k principal components in performing PCA.

   - the first principle component:

   $$w_1 = \arg\max_{\|w\|_2^2 = 1} \; w^T C w$$

   - the second principle component:

   $$w_2 = \arg\max_{\|w\|_2^2 = 1, w_2 \perp w_1} \; w^T C w$$

   equivalently,

   $$w_2 = \arg\max_{\|w\|_2^2 = 1} \; w^T (C - w_1 w_1^T) w$$

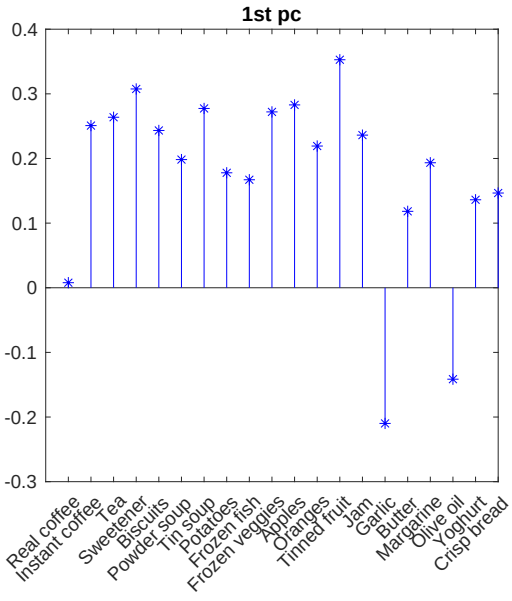
- ...

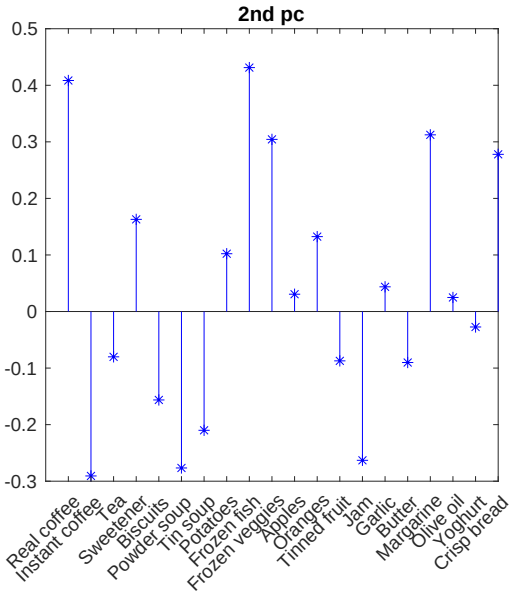
- the kth principle component

$$w_2 = \arg\max_{\|w\|_2^2=1} \; w^T (C - \sum_{i=1}^{k-1} w_i w_i^T) w$$

rubric:
reasonable attempt, 2pt.
Accurate optimization formulation for the first pc, 2pt,
Complete formulation 1pts

3. (7 points) Find the top two principal component vectors for the dataset and plot them (plot a value of the vector as a one-dimensional function). Describe do you see any pattern. You may either use a package or write your own code.

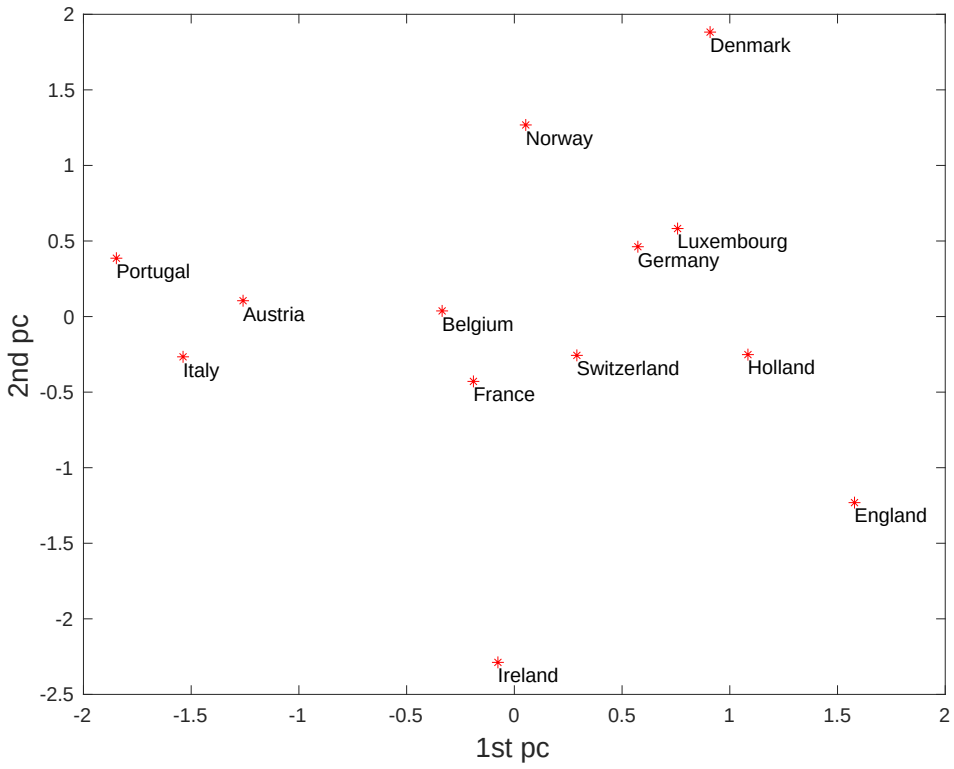




The first two eigenvectors of the covariance matrix, i.e., the top to principle component $w_1, w_2$are shown as the above plots. Each entry of the eigenvector is a weight for the corresponding features.
In the first plot, we can see that most of the entries of w_1 have similar value, this implies that the first eigenvector captures the mean of the data features. For the second eigenvectors w_2, on the contrary, is capturing the difference of the data features, since the entries of the w_2 have opposite signs and different values.

rubric:
each correct pc plot, 3pts.
reasonable explanation 1pt

4. (8 points) Now project each data point using the top two principal component vectors (thus now each data point will be represented using a two-dimensional vector). Draw a scatter plot of two-dimensional reduced representation for each country. What pattern can you observe? You may use use a package or write your own code.



From the scatter plot we can learn that food consumption habits of the people from Danmark, England and Portugal are much more distinctive, since their projected locations are on the edges of the plot. Other counties tend to have more moderate food consumption styles.

rubric:
reasonable attempt 2pts correct scatter plot 5pts. (Note that the submitted plot may have different look due to the orientation difference in the pc definition)
reasonable explanation 2pt