Habib University
CSE 351 - Artificial Intelligence
Fall' 2022
Assignment 03 - Clustering and Recommendation

40 Points

Objective:

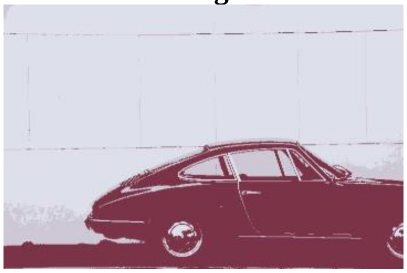
This assignment asks students to implement clustering and recommendation techniques to gain better understanding of their internal working. The students will gain hands-on experience of image segmentation via KMeans and will use gradient descent to perform matrix factorization.

Q 1 – Clustering [20 points]

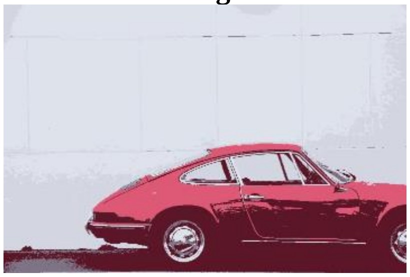
KMeans has wide applications in different areas including image processing. In this question, you will use Kmeans to perform image segmentation which is a process of partitioning an image into multiple regions.Your KMeans implementation will take an image as input and number of clusters (K) and will segment the image into K number of clusters based on the similarity of pixels as shown below:

Original Image

Clustered image with K=3



Clustered image with K=5



The resultant image should ideally show each or similar objects in same cluster. Similarity, in the context of this assignment, is determined by RGB values.

a) **[14 Point]** Implement K-means algorithm to cluster these points. K-means will take an image and number of clusters (K) to be formed and will partition the image into given number of clusters. The algorithm will stop if there is no significant change coming in these segments.

b) **[06 Points]** Provide graphical visualization of the process of formation of clusters.

The code will be written in the attached Q1_clustering.py file. Some sample images are attached with the assignment. In addition to submitting the code, you will also submit the result of clustering of these images in a pdf file.

Q2 - Collaborative Filtering [20 points]

You are implementing collaborative filtering to make recommendations to users U for items I. You decided to use a model-based approach that applies matrix factorization to factorize a rating matrix (R) into User features (P) and Item features (Q) such that,

$$P \times Q \quad = \hat{R} \cong R$$

where P is a $m \times k$ matrix and Q is $k \times n$ matrix and m, n and k represent no. of users, no. of items and no. of latent factors respectively.

The predicted rating of an item *j* by user *i* is calculated as follows:

$$\hat{r}_{ij} = \sum_k p_{ik} q_{kj}$$

and error in this prediction is calculated as:

$$e_{ij}^2 = (r_{ij} - \sum_k p_{ik} q_{kj})^2$$

a) Computing gradients [5 points]

You are applying gradient descent to minimize mean square error. Compute gradient of error with respect to p and q values and derive formulas to update p and q values. The calculations of p would be:

$$\Delta p_{ij} = \frac{\partial e_{ij}^2}{\partial p_{ij}}$$

$$p'_{ij} = p_{ij} - \alpha \, \Delta p_{ij}$$

Where p' is updated value of p and α is learning rate.

b) Adding biases [5 points]

There can be biases in user and item recommendations, which are handled by introducing user bias vector (U) and item bias vector (I). In the presence of these biases, rating r_{ij} is predicted as follows:

$$\hat{r}_{ij} = U_i + I_j + \sum_k p_{ik} q_{kj}$$

Derive formulas to update p, q, and biases during gradient descent.

c) Factorizing matrices using gradient descent [10 points]

In this question, you will implement gradient descent technique to perform matrix factorization. Given a rating matrix, your code will apply matrix factorization to determine User features (P), Item features (Q), User bias vector (U) and Item bias vector (I). These matrices will then be used to make recommendations. The code will be written in the attached Q2_Recommendation.py file.

Submission Instructions

The assignment will be done in pairs. Submissions will be made on Canvas by the due date (announced on Canvas). The submitted file should be in the form of a ZIP file named as **<studentid1>_<studentid2>_Ass3** containing a pdf document (for Q1 results and Q2-a and b) and python code for both Q1 and Q2.