HW# 6

Entrega: Segunda-feira, 7/12/2015 (23:59)

Atenção: Justifique todas suas respostas e faça upload do seu código em formato zip.

1. Exercise 6.11

2. In this exercise you will apply k-means to Image Compression.

   In a straightforward 24-bit color representation of an image, each pixel is represented as three 8-bit unsigned integers (ranging from 0 to 255) that specify the red, green and blue intensity values. This encoding is often referred to as the RGB encoding. Will will work again on image compression on the same image from HW#5. Our image contains thousands of colors, and in this part of the exercise, you will reduce the number of colors to 16 colors. By making this reduction, it is possible to represent (compress) the photo in an efficient way. Specifically, you only need to store the RGB values of the 16 selected colors, and for each pixel in the image you now need to only store the index of the color at that location (where only 4 bits are necessary to represent 16 possibilities).

   (a) Implement Lloyd's algorithm for k-means clustering. Your function should receive as inputs the number of clusters and the dataset and return: (1) the position of the final centroids, (2) the allocation of the data into the clusters, and (3) the final E_{in}.

   (b) Load the Nixon-Presley Meeting image or any other of your preference and then reshape it to create an $m \times 3$ matrix of RBG pixel colors (where $m = width \times height$).

   (c) Call your K-means function on the data with K from 1 up to 16. Plot the $E_{\text{in}} \times K$.

   (d) After finding the top K = 16 colors to represent the image, you can now assign each pixel position to its closest centroid. This allows you to represent the original image using the centroid assignments of each pixel. Notice that you have significantly reduced the number of bits that are required to describe the image. The original image required 24 bits for each one of the $width \times height$ pixel locations, resulting in total size of $width \times height \times 24$ bits. The new representation requires some overhead storage in form of a dictionary of 16 colors, each of which require 24 bits, but the image itself then only requires 4 bits per pixel location. The final number of bits used is therefore $16 \times 24 + width \times height \times 4$ bits. What is the factor of compression for your image?

   (e) Finally, you can view the effects of the compression by reconstructing the image based only on the centroid assignments. Specifically, you can replace each pixel location with the mean of the centroid assigned to it. Display the original and the modified data.

3. The Iris dataset is one of the most popular toy datasets used in classification tasks. It contains the *sepal length*, *sepal width*, *petal length*, and *petal width* of three different types of *Iris flowers*.[1] This dataset is available in the datasets R package, the SciKit learn library for Python, and Matlab. It may also be downloaded from `https://archive.ics.uci.edu/ml/datasets/Iris`.

   In this question you shall apply both clusteting and classification algorithms to the Iris Dataset. First we assume the labels are known.

   (a) Split your dataset into `train` (80%) and `test` (20%).

   (b) Using the `train` dataset, implement the k-fold cross-validation to determine the number of neighbors k in the k-NN algorithm. Plot the cross-validation prediction error for $k = 1, 3, 5, ..., 15$.

   (c) Using the chosen k fit the whole dataset and report the confusion table for the `test` dataset.

   (d) Fit a linear classifier to the `train` dataset and apply the resulting hypothesis to your `test` dataset. Report the confusion table.

   (e) Which classifier works better in this dataset?

   (f) Now we turn to a clustering problem and will use the whole dataset. Assume **the labels are unknown** and your task is to separate the flowers in clusters. You **know there are three types of flower**. Implement a k-means algorithm to cluster your data and, afterwards, use the actual labels to construct the classification errors/confusion table. Note that, in general, it is not possible since the clusters are unknown a priori.

   (g) Now assume that the **number of clusters is unknown**. Implement a 10-fold cross-validation to find the number of clusters in the k-means algorithm with $k = 2, 3, 4, 5, 6, 7$ and report cross-validation fit of your hypothesis for each k.

4. Problem 6.14

[1] https://en.wikipedia.org/wiki/Iris_flower_data_set.