Homework 1

DS 540 Data Mining

Maximum points: 100

Due: Feb 9 (Thursday), 11:59pm

For this assigment you will be computing distance and similarity measures between vectors. Please make sure you code these distance/similarity measures yourself and not get them from some python package.

1: Euclidean/Minkowski distance between real vectors (40 points)

1. (**10 points**) Write a python function called $euclid\_dist(x, y)$ that takes 2 lists x and y and returns the euclidean distance between these 2 lists. For example, if x = [1,2,1] and y = [1,1,1] your function should return the value 1 ($\sqrt{(1-1)^2 + (2-1)^2 + (1-1)^2}$). Here it is required to code the formula and not use some library for computing the distance.

2. (**10 points**) For the following 4 points in 2-dimension: (0,2), (2,0), (3,1) and (5,1), using the $euclid\_dist(x, y)$ function from part 1.1, compute and display the pairwise distances as a numpy array. Hence, your displayed array will have dimensions 4x4.

3. (**20 points**) Generalization of euclidean distance is called Minkowski distance. Here, for given lists x and y and a non-negative parameter r, the distance is defined as:

$$d(x, y) = \Big(\sum_{k=1}^{n} |x_k - y_k|^r \Big)^{1/r} \tag{1}$$

   - (**10 points**) Write a python function $minkowski\_dist(x, y, r)$ that returns the minkowski distance between any 2 lists x and y with a given parameter r.
   - (**5 points**) For r = 0.5 using the function $minkowski\_dist(x, y, r)$ compute and display the pairwise distance for the 4 points in part 1.2. Hence, here again you should display a 4x4 numpy array.
   - (**5 points**) Repeat the above question with r = 2

2: Similarity between binary vectors (20 points)

Write a python function $binary\_sim(x, y)$ that takes 2 lists x and y of binary numbers (containing 0s and 1s) and returns the Similarity matching and Jaccard coefficients for these lists.

3: Correlation and Cosine distances (40 points)

For any 2 given lists x and y of real numbers:

1. (**10 points**) Write a python function $cosine\_dist(x, y)$ that computes and returns the cosine distance between x and y
2. (**30 points**) Write a python function $corr\_dist(x, y)$ that:
   (a) (**10 points**) Computes the covariance between x and y (cov).
   (b) (**10 points**) Computes the standard deviation of x and y ($sigx$ and $sigy$ respectively).
   (c) (**5 points**) Computes the correlation between x and y (corr)
   (d) (**5 points**) returns the following quantities as a list of 4 numbers: $[cov, sigx, sigy, corr]$.