Project to implement mCLESS algorithm and experiment on various datasets

Creating the mcless module

Information matrix

Create a module named Infomatrix which contains 2 files:

1. __init__.py
2. Information_matrix.py

In Information_matrix we will write the code to augument the data matrix by columns of 1s in the left side of the matrix

Source Matrix

Create a module named srcmatrix which contains 2 files:

1. __init__.py
2. Source_matrix.py

In the Source_matrix we will write the code to get the source matrix from the labels of the data.

mcless

Create a module named mcless which contains 3 files:

1. __init__.py
2. Calculate_W.py
3. mCLESS.py

Calculate_W.py

The least squares method is formulated as

$$\widehat{W} = \underset{W}{\operatorname{argmin}} ||AW - B||$$

I wrote 2 functions in Calculate_W.py

1. calculate_W_by_svd()

Using Singular Value Decomposition, the least sqaures solution is

$$x = \sum_{i=1}^{r} \frac{u_i b}{\sigma_i} v_i$$

In Calculate_w.py , we calculte the weight matrix using SVD for least squares method which is formulated as

We can decompose $A = U\Sigma V^T$. So, we get W using the above method as,

$$W = \sum_{i=1}^{r} \frac{u_i b}{\sigma_i} v_i$$

2. calculate_W_by_normal()

This function calculates W by method of normal equations

$$(A^T A)\widehat{W} = A^T B$$

So

$$\widehat{W} = (A^T A)^{-1} A^T B$$

Since the solutions by both methods were the same for all the datasets in this experiments, only calculate_W_by_normal() is used for all experiments.

mCLESS.py

This file contains the fit(), predict(), and score() methods for the mCLESS class. We give our train set to fit() method to calculte the $\widehat{W}$. Then the classes are predicted using the predict() method for the test set. score()

Following Documents are the attached codes for the above mentioned files :

In [ ]: