Homework 1 Optimization for data science

Rasetta Adriano Rossetto Alberto

1 Introduction

The following homework focuses on a semi-supervised learning task. This type of task includes both labeled and unlabeled samples. The goal is to assign the correct label to unlabeled samples. This kind of problem is very close to real-world scenarios where you have a lot of data but few labels (unlabeled data). The paradigm of reference for the resolution of the task was "similar features share similar labels".

2 Method

The following considerations on the methods adopted were initially applied to an artificial dataset and then to a real one. The optimization problem to be solved is the following:

$$\min_{y \in R^{\mu}} f(y) = \min_{y \in R^{\mu}} \sum_{i=1}^{l} \sum_{j=1}^{\mu} w_{ij}(y^j - \bar{y}^i)^2 + \frac{1}{2} \sum_{i=1}^{\mu} \sum_{j=1}^{\mu} \bar{w}_{ij}(y^j - y^i)^2$$

We define two similarity matrices: the first contains distances between labeled and unlabeled samples and the second distances between unlabeled samples. For an appropriate resolution of the problem, a modified version of the Euclidean distance was chosen as a measure of similarity.

$$d(x_1, x_2) = \frac{1}{\sqrt{\sum_{i=1}^{n}(x_{1i} - x_{2i})^2} + 1}$$

With the classical Euclidean distance, very close (similar) points have a low distance. With this modified version there is a direct proportionality: similar points have high similarity values (maximum value is one), distant points have low values.
Three algorithms have been implemented: the classic gradient descent and two block coordinate gradient descent methods (cyclic and randomized) with blocks of dimension equal to one. The following formula was used to calculate the gradient:

$$\frac{\partial f(y)}{\partial y^j} = 2 \sum_{i=1}^{l} w_{ij}(y^j - \bar{y}^i) + 2 \sum_{i=1}^{\mu} \bar{w}_{ij}(y^j - y^i)$$

Through some experiments we observed little differences in performance using different stepsizes between the three algorithms. For this reason the same stepsize has been applied in all three algorithms. In order to find the most appropriate stepsize we perform a grid search using the gradient method with different values for the stepsize, this procedure was applied for each dataset.
A maximum number of iterations was set for each algorithm: in the case of the gradient descent and the cyclic BCGD the same number of iterations were used, while for the randomized BCGD it was necessary to use a higher number of iterations to achieve convergence.
An additional stopping criterion was used for each algorithm. Although for computational reasons it was not possible to calculate the value of the function at each iteration, a minimum threshold was set between the last two function values calculated. Through experiments an appropriate threshold was set for each dataset.

$$|f(y)_{k-1} - f(y)_k| < threshold$$

3 Artificial dataset

The artificial dataset was generated randomly from a Gaussian distribution and included 1000 samples and 2 features with only 10% samples labeled.

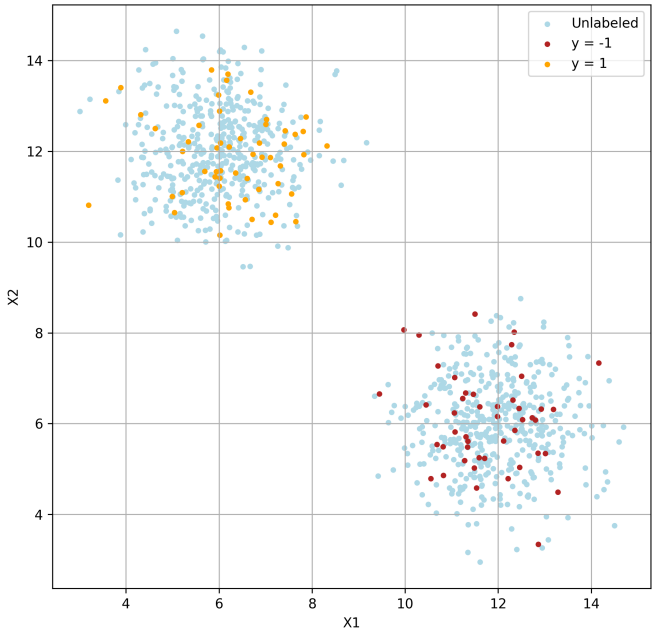

Figure 1: Artificial dataset.

Regarding the number of iterations, the randomized BCGD requires a very high number of iterations to converge with respect to the other two algorithms. From the plot is possible to observe how the three algorithms have similar performances in terms of accuracy with respect to the execution time.

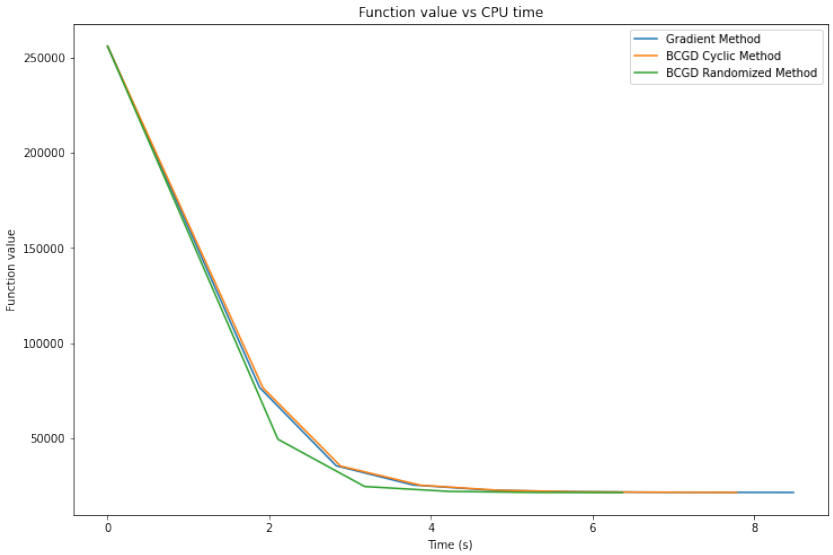

Figure 2: Comparison between algorithms in the artificial dataset.

4 Real dataset

4.1 Description

In the following part we applied the optimization algorithms to a real dataset downloaded from the UCI Machine Learning Repository. The dataset describes two species of rice (Osmancik and Cammeo) and is constitued by 3810 instances and 7 features:

- Area;
- Perimeter;
- Major Axis Length;
- Minor Axis Length;
- Eccentricity;
- Extent.

In order to make suitable the dataset for the procedure we applied a transformation of the response variable in -1,1.

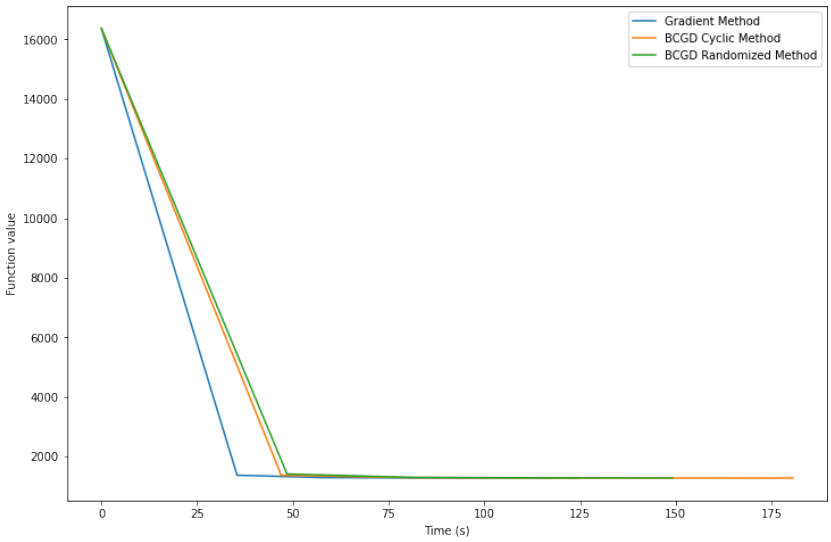

Figure 3: Comparison between algorithms in the real dataset.