

Troy Daniels

Machine Learning Foundations Lab

12/01/2023

This is a lab experiment comparing the accuracy and computational complexities of simple learning algorithms (classification). Given a two dimensional (input vector of size two) linearly separable dataset, how well does gradient descent (using various loss functions) learn to successfully classify the data and what is the computational complexity required to reach such success. Additionally, does a simple perceptron algorithm when given the same dataset and goal have better results?

The dataset is created by `#MachineLearningLab.generateData(dim=2, size=100)` and is set to create a random dataset (of a size 100) where each set of data is two dimensional and each number exists in the range (0,5). Each piece of data in the dataset is given a label (either 0 or 1) which is determined by a random line separating the data which exists within the two dimensional space of the dataset.

The purpose of this lab compares the speed, number of iterations and accuracy of these algorithms in learning to successfully classify the linear separable data. The algorithms compared are as follows:

1. Gradient decent - least squared loss
2. Gradient decent - crossEntropyLoss
3. Gradient decent - softmax loss
4. Perceptron learning algorithm (basic)
5. Perceptron learning algorithm
6. Linear programming algorithm

The algorithms first trained on a randomly generated linearly separable dataset of size 100, separated by line Y. The number of iterations and time to train are recorded. Another randomly generated linearly separable dataset of size 100, separated by the same line Y, is used to test how well the algorithm generalizes its predictions by recording the number of incorrect predictions on this dataset. This process is repeated 100 times and results are averaged out.

Results:

Algorithm	Incorrect	Iterations	Runtime
least squared loss	2.29	2098	11
crossEntropyLoss	23.85	1982	9
softmax loss	1.73	1903	4
Perceptron (basic)	1.4	50	Near 0
Perceptron	1.4	50	Near 0
Linear programming	1.23	NA	Near 0

The least squared loss algorithm on average got 2.29 incorrect. It took 2098 iterations and 11.0 time to train

The cross entropy loss (GD) on average got 23.85 incorrect. It took 1982 iterations and 9.0 time to train

The soft max loss (GD) on average got 1.73 incorrect. It took 1903 iterations and 4.0 time to train

The basic perceptron algorithm on average got 1.4 incorrect. It took 50 iterations and 0.0 time to train

The perceptron algorithm on average got 1.4 incorrect. It took 50 iterations and 0.0 time to train

The linear programming algorithm on average got 1.23 incorrect. It took 0.0 time to train



