# Direction-Boundary Set Reduction

Chantz T. Large
University of Colorado – Colorado Springs

***Abstract*—Reducing the size of the dataset is of interest to those working with resource constrained devices or performing incremental learning. This paper introduces a new direction-boundary Set Reduction strategy for reducing the size of the dataset. The algorithm in its current state performs well for problems involving large-scale sparse datasets which are reasonably linearly separable. Executed on the popular letter scale data set, it has been shown to reduce the size of the dataset by 37% while maintaining its model generating integrity (2% accuracy reduction).**

## I. Introduction

The direction-boundary algorithm works well for point-preservation between phases of incremental learning, as well as, dataset maintenance on resource constrained devices. The algorithm is designed to be simple, lightweight and easily-extensible. At its root, the algorithm is design to identify the relevant boundary points of the dataset; preserving only the points necessary for defining the shape of the set itself. In its current state, the algorithm has only been developed with datasets involving convex classes which are generally linearly separable.

Applications for this algorithm extend to those involving; incremental learning where maintenance of the training batch between training phases is unreasonable, resource constrained devices where storage capacity or available memory is scarce, or where reducing the training time of the dataset itself is of concern.

Many current machine learning algorithms share much consideration for the amount of resources necessary to conduct learning, or are constructed on the premise that the training batch will be maintained between training phases. While concurrent algorithms continue to elevate the computational bounds for unprecedented modeling of large datasets, many of the previously mentioned applications are unable to capitalize on these innovations. This work in particular was motivated by the incremental learning library, LIBLINEAR.

## II. method

The method for point-selection involves the following steps; projection and fitting of the direction vectors to the original dataset, identifying points residing nearest to the projected boundary, Fig. 1.

Projection of the boundary involves calculating the direction vectors defined by the mean center of the dataset and a corresponding point within the dataset. After all points have been projected the boundary is then fitted to the original dataset by scaling by the maximum and minimum values in all dimensions. Finally, distance is measured and points nearest to the projected boundary are retained.

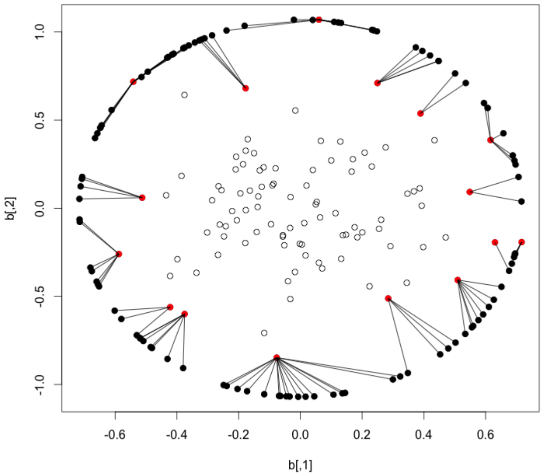

Fig. 1. 2-dimensional plot illustrating point-identification; empty circles represent original dataset, filled circles represent projected boundary, filled-red circles represent selected points. Lines to and from boundary points to selected points illustrate voting.

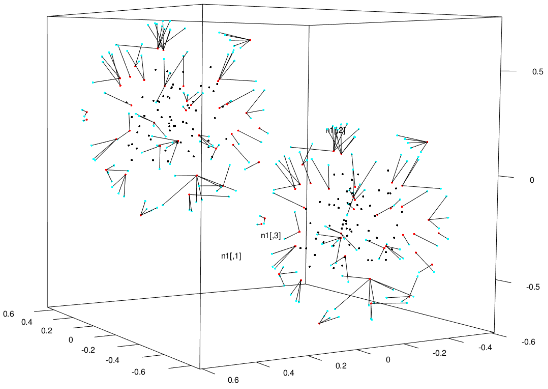

Fig. 2.

## III. evaluation

Early evaluation of the direction-boundary algorithm demonstrates promising results. Preprocessing of the letter training set effectively reduced the size of the set by 37% and while retaining acceptable model generating integrity (2% accuracy loss). As the scale of the data set increases, the performance of the algorithm improves, Fig. 5.

For illustrative purposes, the algorithm has been demonstrated to be effective on skewed datasets as well, Fig. 4.

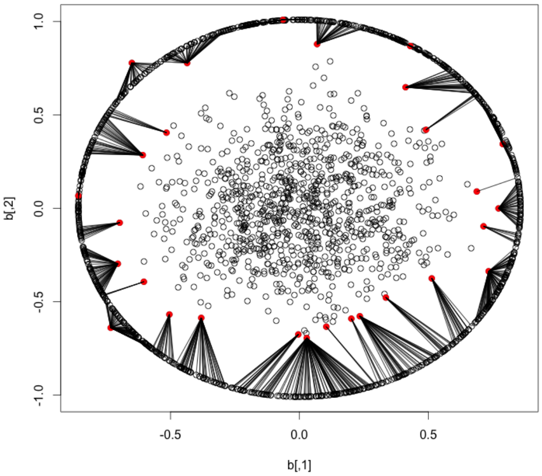

Fig. 3.

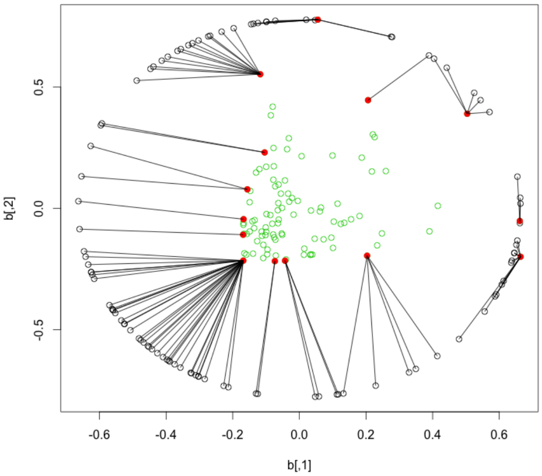

Fig. 4.

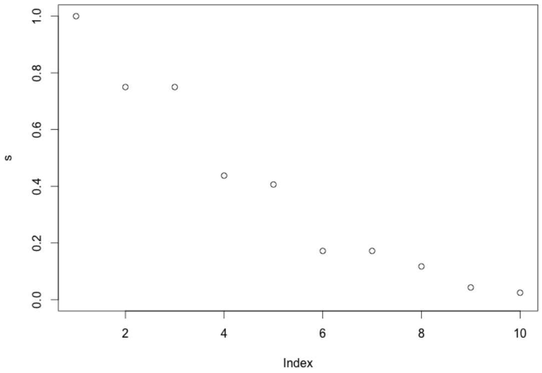

Fig. 5.