Data Expressiveness and Its use in Data-centric AI

Hasan Kurban, Parichit Sharma, Mehmet Dalkilic
Computer Science/Data Science
Indiana University, Bloomington

Abstract

To deal with the unimaginable continual growth of data and the focus on its use rather than its governance, the value of data has begun to deteriorate seen in lack of reproducibility, validity, provenance, etc. In this work, we aim to simply understand what is the value of data and how this basic understanding might affect existing AI algorithms, in particular, EM-T(traditional expectation maximization) used in soft clustering and EM* (a data-centric extension of EM-T). We have discovered that the value of data--or its "***expressiveness***" as we call it--is procedurally determined and runs the gamut from ***low expressiveness (LE) to high expressiveness (HE)***, the former not affecting the objective function much, while the latter a great deal. By using balanced binary search trees (BST) (complete orders) introduced here, we have improved on our earlier work that utilized heaps (partial orders) to separate LE from HE data. EM-DC (expectation maximization-data centric) significantly improve the performance of EM-T on big data. EM-DC is an improvement over EM* by allowing more efficient identification of LE/HE data and its placement in the BST. Outcomes of this, aside from significant reduction in run-time over EM*, while maintaining EM-T accuracy, include being able to isolate noisy data, convergence on data structures (using Hamming distance) rather than real-values, and the ability for the user to dictate the relative mixture of LE/HE acceptable for the run.

Background

More than 50 years ago the well-known physicist Feynman observed what he believed to be a looming problem which we paraphrase here: as computing technology advances, the ratio of time devoted to computing over the data to the time required to move data will tend toward zero [1].

Big data, and its now typical pairing with the popular area of data science, has exposed an obvious weakness of iterative algorithms--while the run-time of visiting the data is O(n), as n routinely runs into sizes of 100s of gigabytes and even several terabytes (and, in the near future petabytes) and dimensions regularly spanning 100s into 1000s of features, the operation of visiting each and every datum overwhelms many traditional iterative algorithms.

Introduction

The contribution of the work presented here is to:

1. Refine our approach to determining value of data which we call ***data expressiveness*** whose semantics is determined procedurally.

2. Explore the potential of partial and complete order structures to identify and demarcate the low and high expressive search space.

3. Apply this characterization using balanced binary search trees (BST) to improve a popular AI algorithm-expectation maximization (EM) which has had an earlier data-centric improvement using heaps called EM* [2, 3]. We call this improved data-centric version EM-DC (EM with data-centrism).

Dataset

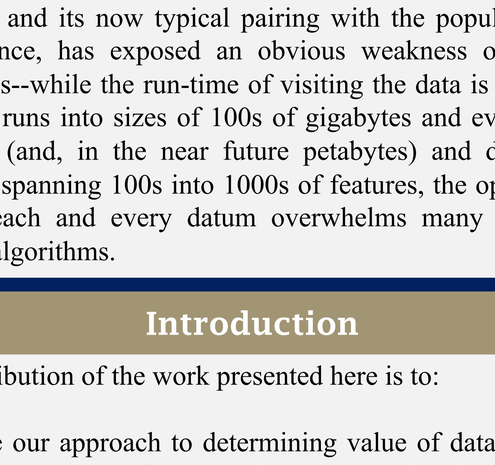


Figure-1: Cropland mapping dataset

We used the cropland mapping dataset [4]. The classification task is to predict the label (i.e., type of the crop) for a given data point. There are 325,834 records in 175 dimensions and 7 classes. To balance the class distribution, we only consider the 5 dominant that account for ~97% of the data.

Data Centric Technology (brief overview)

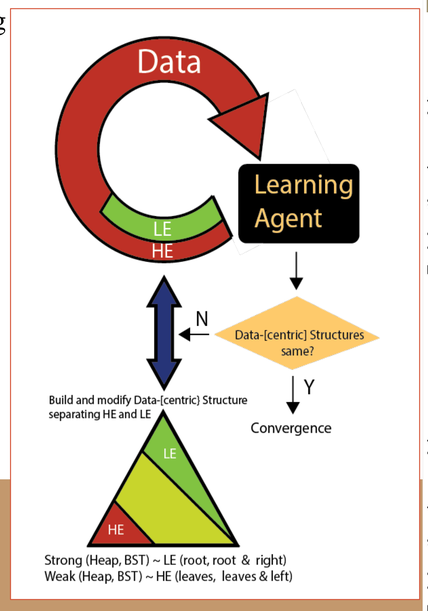
Data Expressiveness: Value of data to learning agent (LA) as a function of location in loop (and other data)

High Expressive Data (HE) affects ↑LA
Low Expressive Data (LE) affects ↓LA

Data-[centric] Structures Heap, BST: Strong regions hold HE, weak regions hold LE.

Iteration works either LE + HE or HE (heuristic alternating).

Convergence is when Data-[centric] structure HE regions do not change significantly.



Representative Example

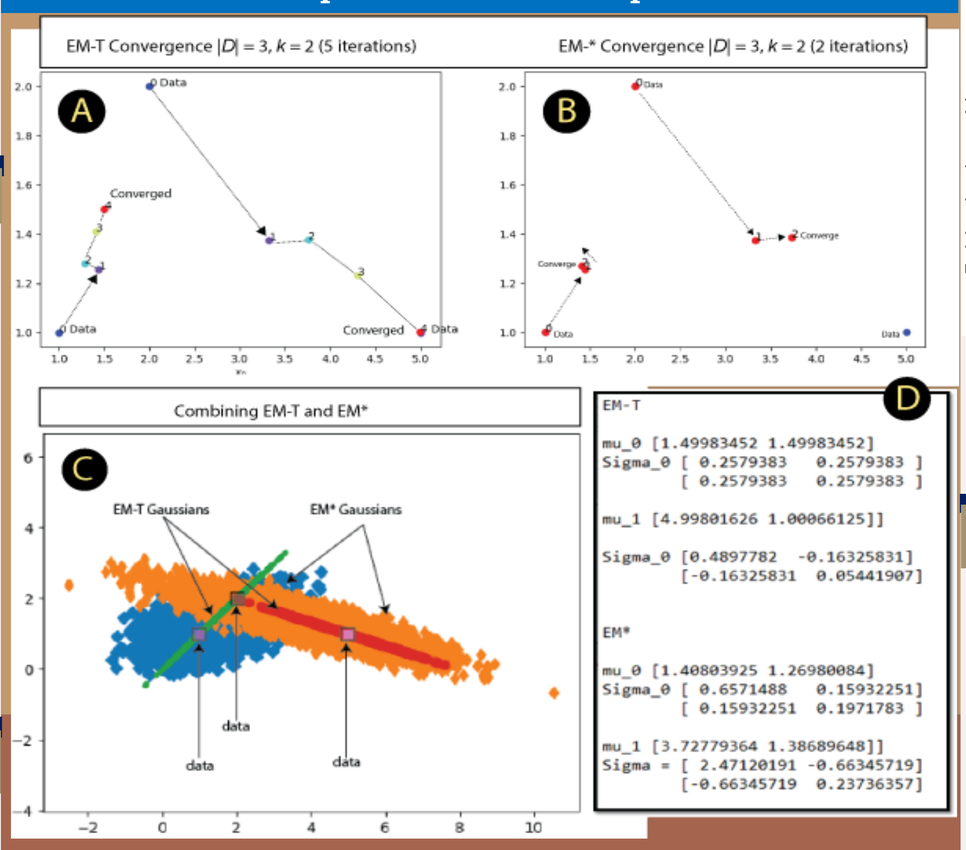


Figure-2: EM-T versus (B) data-centric EM* with a small data set. Each arrow is a step. The iterations for EM* are about 1/2 of EM-T. (C) is an overlay of both showing the variance, data points, and paths toward convergence. (D) shows actual meu, Sigma with meu between EM-T and EM* being unexpectedly close with disparate iterations.

Experiments

A: Clustering Experiments (comparison of accuracy via Adjusted Rand Index, 0: random clustering, 1: perfect clustering compared to ground truth)

Number of clusters	EM-T (seconds)	EM*	EM-DC
5	**0.53**	**0.6**	**0.6**
10	0.62	0.6	0.62
20	**0.71**	**0.72**	**0.71**
30	**0.73**	**0.73**	**0.7**
35	0.76	0.74	0.74

B: Dimensionality Experiments (comparison of execution time)

Number of dimensions	EM-T	EM*	EM-DC
10	**69**	**27**	**12**
20	**78**	**35**	**17**
50	**58**	**34**	**34**
80	**139**	**30**	**18**
100	**70**	**26**	**15**

C: Scalability Experiments (comparison of execution time)

Data Size (# of data points)	EM-T (seconds)	EM* (seconds)	EM-DC (seconds)
100, 000	**75**	**27**	**22**
150, 000	100	44	43
200, 000	**93**	**66**	**47**
250, 000	**103**	**108**	**67**
320, 000	**178**	**96**	**74**

Table-1: EM-DC significantly outperform EM-T and EM* for execution time and number of iterations. Accuracy wise, EM-DC either improves or preserves the accuracy compared to EM-T and EM*.

Results

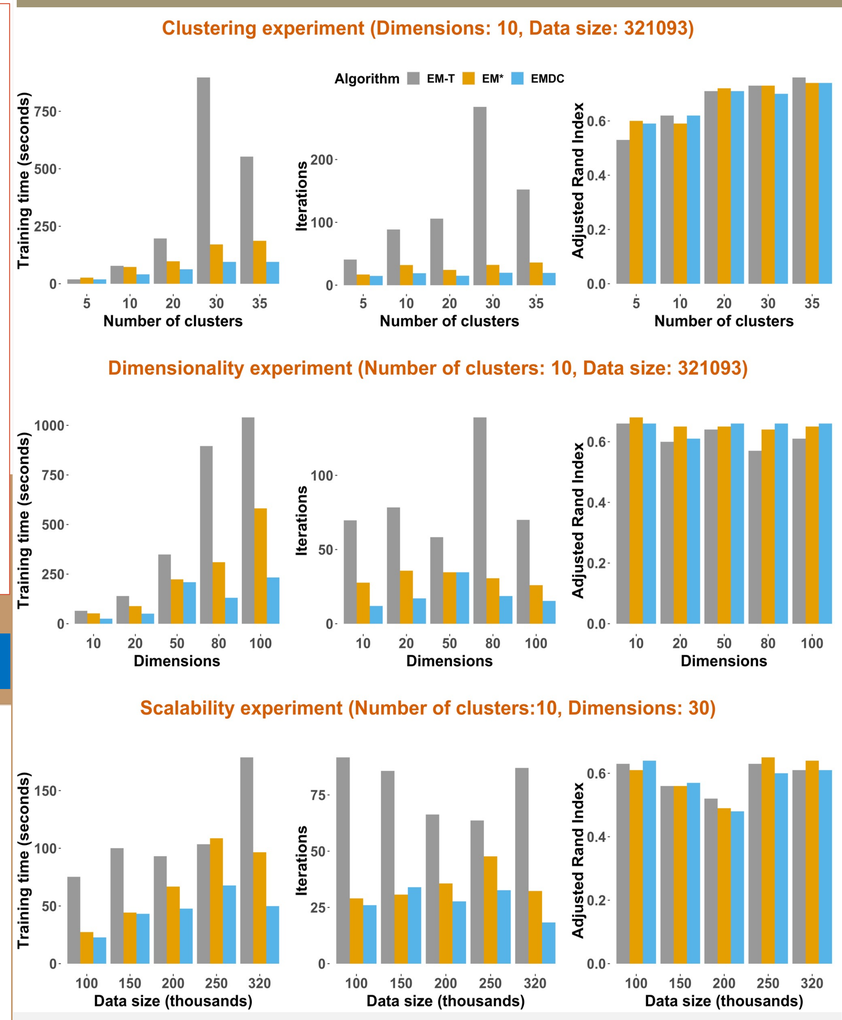


Figure-3: Comparison of EM vs. data-centric EM on real-world data across different aspects. EM-DC is noticeably faster than EM-T and EM* regardless of the number of clusters, feature size or data size. In general, data-centric AI outperforms its non-data-centric counterpart.

Discussion

Experimental findings show that EM-DC is as good or significantly better than both EM-T and EM* (Table-1 and Figure-3).

- The performance gap widens with increment in the number of clusters, dimensions and data size.
- For training time and number of iterations, EM-DC significantly outperforms both EM-T and EM*.
- In terms of classification accuracy (adjusted rand index), all algorithms perform similar.

It should be noted that considerable reduction in total training time and number of iterations is achieved by EM-DC while using only a **third of the data** in the structure i.e., balanced binary search tree whereas, EM* use data in all the leaf nodes (approx. half of the data) and EM-T uses all the data.

Summary

- ***Data Expressiveness & Data-[centric] Structures*** exist and be leveraged to improve data-centric AI algorithms [iterative optimization].
- ***Data Expressiveness*** is latent, intrinsic, variable depending on the entire corpus of data as well as the learning agent depending on when and how data is used.
- ***Data-[centric] structures*** uniquely capture data expressiveness, e.g., strong and weak heaps, that now become vehicle of convergence (or learning).

We use Data-expressiveness and Data-[centric] structures to significantly improve run-time of Expectation-Maximization, first heaps, then in this work BST.

References

[1] R.P. Feynman.Lectures on Computation (Frontiers in Physics) 1st Ed.CRC Press, 2000.
[2] Sharma Parichit, Kurban Hasan, Jenne Mark and Dalkilic Mehmet (2020). DCEM: Clustering Big Data using Expectation Maximization Star (EM*) Algorithm. R package version 2.0.4. https://CRAN.R-project.org/package=DCEM
[3] H. Kurban, M. Jenne, and M.M. Dalkilic. Using Data to Build a Better EM: EM* for Big Data.International Journal of Data Science and Analytics, 4(2):83–97, 2017.
[4] Iman Khosravi and Seyed Kazem Alavipanah. A random forest-based framework for cropmapping using temporal, spectral, textural and polarimetric observations.International Journalof Remote Sensing, 40(18):7221–7251, 2019. doi: 10.1080/01431161.2019.1601285.

Acknowledgement

We appreciate the technical support extended by Rob Henderson (Luddy School of Informatics, Computing & Engineering) in creating the compute infrastructure for this work.

LUDDY