Streaming algorithms for MEB under the Bregman divergences

Alexander Clemmer, Amirali Abdullah , John Moeller

December 5, 2011

1 Introduction

Given a point set P, and a distance function D, the Minimum-Enclosing-Ball (MEB) problem is defined as follows: Find $\min_{c,r} D(p,c)$ s.t. $p \leq r$ for every $p \in P$. The basic purpose of an MEB is to calculate *extent* of a data set. Additionally, the center of the MEB so located, is the point s.t maximal distortion over all elements of P is minimized. We consider the MEB ball under the *Bregman divergence D_ϕ*, that is:

$$D_\phi(x,y) = \phi(x) - \phi(y) - \langle \nabla \phi(y), x - y \rangle \tag{1.1}$$

We define our MEB balls to be left balls, i.e, the center is always the second argument of D_ϕ. This has the benefit that left-balls are always convex, as the Bregman divergence is convex in it's first argument, and that certain intermediate minimization problems become simpler, as we shall see later.

Although our analysis in this report extends to any Bregman divergence, we run our experiments on the *KL*-divergence, defined as :

$$D_{\mathrm{kl}}(a,b) = \sum_{i=1}^{d} a_i \log \frac{a_i}{b_i} - (a_i - b_i) \tag{1.2}$$

We use the *KL*-divergence because it is perhaps the most important Bregman divergence in practice, is easy to compute analytically, is convex in both arguments, and also has the most common challenging properties in analyzing Bregman diverences, i.e, distances and curvature going to infinity sharply (near the axes).

2 Related work

For P in general position, and D is the Euclidean distance l_2, there always exist some $d + 1$ points that support the MEB of P. This is then a discrete problem, and can naively be solved in $O(n^{d+1})$ time. An exact solution can also be found in expected linear time, as first demonstrated by Welzl [6]. The problem can also be cast as one of approximation, and it is known that there exist core-sets for the Euclidean case as shown by Badoiu and Clarkson [2].

The streaming setting denotes that we are constrained to space independent of n, and that we can only read our points sequentially (for the sake of this report, we only consider single-pass algorithms). The problem cannot be solved exactly in this setting, and so we try to find approximation algorithms, i.e, given some error parameter ε , find a ball $B(c,r)$ s.t $D(p,c) \leq r, \forall p$ and $r \leq (1 + \varepsilon)r_{\mathrm{opt}}$. Chan and Zarrabi-Zadeh give a simple streaming algorithm that obtains a 1.5 approximate answer with $O(d)$ space ([7]). Agarwal and Sharathkumar present a somewhat more involved algorithm with $\varepsilon > 0.22$ approximation that uses $O(d \frac{1}{\varepsilon^3} \log \frac{1}{\varepsilon})$ storage ([1]). In our heuristic analysis, we will essentially be considering the analog of Chan's streaming algorithm, and its generalization for Bregman divergences.

We note that the smallest enclosing ball probelm has been considered for Bregman divergences, albeit not in a streaming setting and without theoretically guaranteed convergence time. Nock and Nielsen present algorithms for generalizing both the Welzl algorithm [4] as well as that of Badoui Clarkson [5], and in fact we implement the latter result to give us our baseline estimate of the optimal radius MEB.

3 Main algorithms

We implemented three algorithms. One is the non-streaming base-line estimate of exact MEB derived from the paper of Nock and Nielsen [5]. This was important to do so as to allow us to comapre the performance of our streaming analysis.

Algorithm 1 Nielsen & Nock Algorithm for divergences convex in both arguments

---

    n points, $p_1 \ldots p_n$, $t = T$
    Read point $c = p_i$ at random.
    repeat
        Let s be farthest point from c
        $c \leftarrow \frac{t}{t+1} c + \frac{1}{t+1} s$
    until $t = 0$
    $r = d(p_i, c)$ for p_i farthest from c
    Return c, r

The second one is the most trivial algorithm, which simply takes first point as center and extends it at each step.

Algorithm 2 Find MEB trivial

    Read a stream of n points, $p_1 \ldots p_n$.
    Let $c = p_1$, $r = 0$, $i = 2$
    repeat
        Read point p_i from stream.
        if $D_\phi(p_i, c) > r$ **then**
            Let $r = D_\phi(p_i, c)$
        end if
        $i = i + 1$
    until $i > n$
    Return c, r

Note that algorithm 2 guarantees us a 2 approx for the Euclidean case by the triangle inequality which is often a reasonable ball-park estimate for many applications. However, one would expect this to be poor for Bregman divergences where the triangle inequality may be violated to an arbitrarily bad degree, and in our results we shall demonstrate that this is indeed the case.

The third one is the generalization of the algorithm given by Chan and Zarrabi-Zadeh [7]. We note that this is not immediate, and is the main theoretical contribution of our report.

Note that the key step here is given a $B(c, r)$ and a new point p_i that lies outside of it, we need to find an extension of the ball that contains both the point p AND the original ball. In our next section, we descibe how to do this.

4 Extending ball

We first note given a left-ball $B(c, r)$ and point p, how to find $\max_{x \in B(c,r)} D_\phi(x, p)$. Note that since the Bregman divergence is convex in the first argument, this is a convex optimization problem. We can set up the Lagrange multipliers as done by Cayton (see [3] for details) and a simple computation shows that the solution of the nearest and farthest point in $B(c, r)$ to p is satisfied by x s.t. $D_\phi(x, c) = r$, $\nabla \phi(x) =$

Algorithm 3 Find MEB Ext-ball
_____

Read a stream of n points, $p_1 \ldots p_n$.
Let $c = p_1, r = 0, i = 2$
repeat
    Read point p_i from stream.
    if $D_\phi(p_i, c) > r$ **then**
        Find $c_{\text{new}}, r_{\text{new}}$ s.t $B(c, r) \subset B(c_{\text{new}}, r)$
        $c = c_{\text{new}}, r = r_{\text{new}}$
    end if
    $i = i + 1$
until $i > n$
Return c, r
_____

$\theta \nabla \phi(c) + (1 - \theta) \nabla \phi(p)$. That is, the farthest point (p_{far}) and the nearest point (p_{near}) both lie along the straight line joining c and p in the gradient space.

This immediately suggests a straightforward algorithm for computing c_{new} - simply do a binary search along the straight line joining $\nabla \phi(c)$ and $\nabla \phi(p)$, and find c_{new} s.t $D_\phi(p_{\text{far}}, c_{\text{new}}) = D_\phi(p, c_{\text{new}})$. Although we do not have a proof that our extend ball approach is indeed the minimal radius new ball that encloses both $B(c, r)$ and p, our experimental results support the conjecture strongly.

5 Experimental Results

We run our data over 3 types of point sets. The first group of data sets is a points spread exponentially far apart, and at growing rate. We observe that Algorithm 2 performs poorly as expected and indeed tends to arbitrarily bad approximation ratios as we increase the exponential gap of our points. However, our Algorithm 3 performs fine and matches the exact algorithm 1. We plot the following graph where each point represents the approximation quality of a data set, and increasing x-coordinate indicates greater exponential spread of a data set.
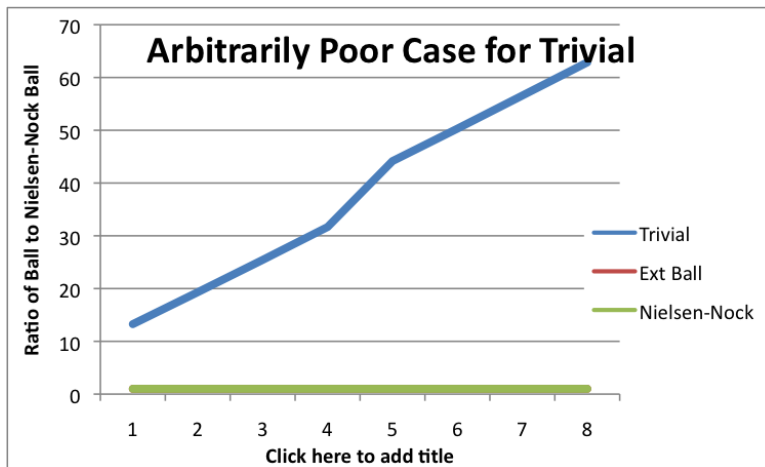

Figure 1: Note the poor performance of the trivial algorithm

For our second type of data set, we pick points regularly spread over the unit hyper cube (in the first

orthant) . Due to numerical issues and overflows, we made sure the points were bounded away from the axes which is where the *kl*-divergence tends to infinity. We found that for these types of data sets , increasing *d* has little to no impact on the quality of approximation obtained. This is not surprising, in the light of [7], in which the worst case approximation bounds do not involve *d*.
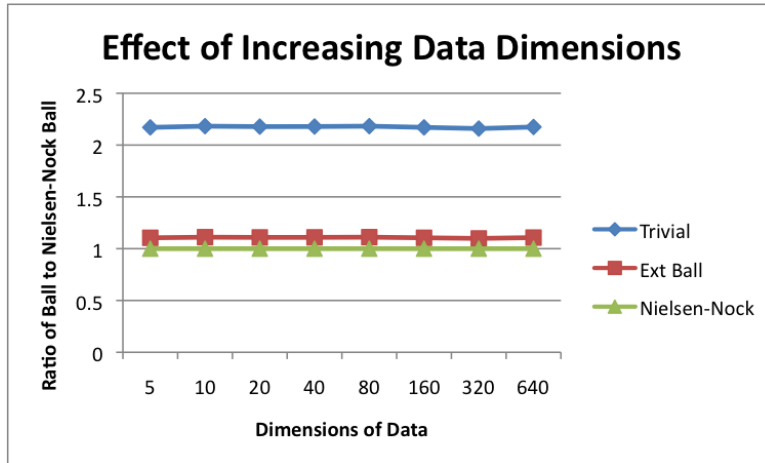

Figure 2: Note the essentially flat performance with increasing dimensionality

Finally, we considered our third type of data set, points spread uniformly around the boundary of a ball. This is the point set that realizes the 1.5 lower bound for the Euclidean case ([7]), and indeed realizes our worst results for Algorithm 3 of 1.6. We obtained this for points spread around the boundary of a Bregman ball, both over a small domain (where a ball is more roughly spherical) and also near the an axis (where a ball is much more anisotropic and elliptical. In both cases, surprisingly, our Algorithm 3 attained a 1.6 bound. It remains open whether taking more anomalous balls or regions will exhibit an arbitrarily bad bound for Algorithm 3 on the lines of Algorithm 2 or whether there is indeed a worst case performance bound.
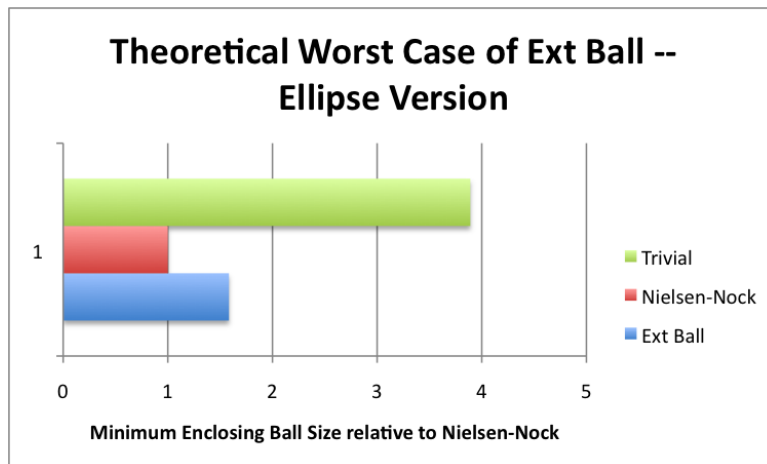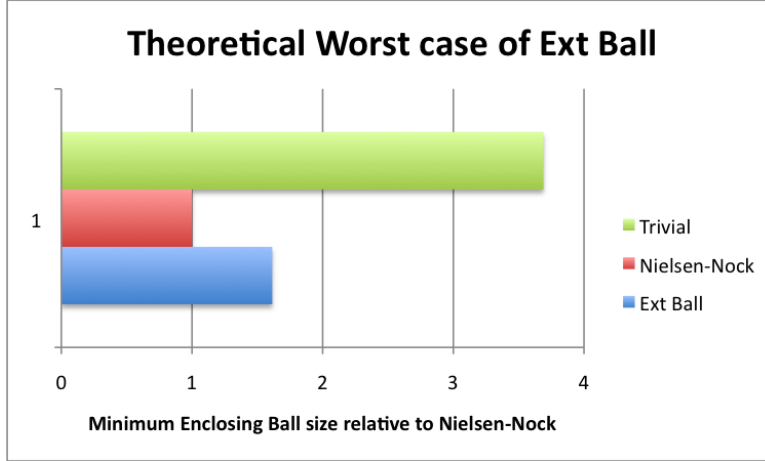

Figure 3: A not too irregular Bregman ball

4

Figure 4: A highly anisotropic Bregman ball

6   Further work

Agarwal and Sharathkumar obtain better approximation guarantees than the ball extension of Chan and Zarrabi-Zadeh ([7]) by essentially maintaining multiple coresets of the data. Before any further such extension, one key question is whether coresets even exist for the MEB under Bregman divergences. The conjecture in the theoretical community is that there is not, without a boundedness assumption, but there is no published result verifying the same.

Secondly, can we give theoretical upper bounds on the approximation quality of our heuristic in a bounded domain, using the fact that a Bregman diveregnce satisfies approximate triangle inequality over any bounded domain?

Thirdly, can we show a higher lower bound on approximation quality than the 1.22 that holds for the Euclidean case if we allow unbounded domain?

References

[1] AGARWAL, P. K., AND SHARATHKUMAR, R. Streaming algorithms for extent problems in high dimensions. In *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms* (Philadelphia, PA, USA, 2010), SODA '10, Society for Industrial and Applied Mathematics, pp. 1481–1489.

[2] BADOIU, M., AND CLARKSON, K. L. Optimal core-sets for balls. *Comput. Geom. Theory Appl. 40* (May 2008), 14–22.

[3] CAYTON, L. *Bregman proximity search*. PhD thesis, University of California, San Diego, 2009.

[4] NIELSEN, F., AND NOCK, R. On the smallest enclosing information disk. *Information Processing Letters 105*, 3 (2008), 93 – 97.

[5] NOCK, R., AND NIELSEN, F. Fitting the smallest enclosing bregman ball. In *Machine Learning: ECML 2005*, J. Gama, R. Camacho, P. Brazdil, A. Jorge, and L. Torgo, Eds., vol. 3720 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 2005, pp. 649–656.

[6] WELZL, E. Smallest enclosing disks (balls and ellipsoids). In *New Results and New Trends in Computer Science*, H. Maurer, Ed., vol. 555 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 1991, pp. 359–370. 10.1007/BFb0038202.

[7] ZARRABI-ZADEH, H., AND CHAN, T. M. A simple streaming algorithm for minimum enclosing balls. In *CCCG* (2006).

A   Distribution of work

John implemented the base-line algorithm 1 for computing exact MEB ([5]) as well as offer several helpful ideas and suggestions for the theory component of this report.

Alexander implemented our own algorithm 3, and ran the major experiments on our data sets as well as handling multiple tricky numerical and computational issues that arise at boundary cases.

Amir handled the major part of theoretical content and background, including the idea on how to extend the ball and construction of the datasets.