Coresets for Polytope Distance[*]

Bernd Gärtner
Institute of Theoretical Computer Science
ETH Zurich, Switzerland
gaertner@inf.ethz.ch

Martin Jaggi
Institute of Theoretical Computer Science
ETH Zurich, Switzerland
jaggi@inf.ethz.ch

ABSTRACT

Following recent work of Clarkson, we translate the coreset framework to the problems of finding the point closest to the origin inside a polytope, finding the shortest distance between two polytopes, Perceptrons, and soft- as well as hard-margin Support Vector Machines (SVM). We prove asymptotically matching upper and lower bounds on the size of coresets, stating that ϵ-coresets of size $\lceil(1 + o(1))E^*/\epsilon\rceil$ do always exist as $\epsilon \to 0$, and that this is best possible. The crucial quantity E^* is what we call the excentricity of a polytope, or a pair of polytopes.

Additionally, we prove linear convergence speed of Gilbert's algorithm, one of the earliest known approximation algorithms for polytope distance, and generalize both the algorithm and the proof to the two polytope case.

Interestingly, our coreset bounds also imply that we can for the first time prove matching upper and lower bounds for the sparsity of Perceptron and SVM solutions.

Categories and Subject Descriptors

F.2.2 [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Problems—*Geometrical problems and computations, Pattern matching*; I.5.1 [**Pattern Recognition**]: Models—*Geometric*

General Terms

Algorithms, Theory

Keywords

Approximation Algorithms, Coresets, Polytope Distance, Support Vector Machines, Kernel Methods, Sparsity, Geometric Optimization

1. INTRODUCTION

Coresets.

The concept of coresets has proven to be a very successful one for approximation algorithms for many discrete geometric problems. On one hand coreset algorithms are much faster than exact algorithms, and on the other hand they simultaneously ensure that the obtained approximate solutions still have very compact (sparse) representations, making them very appealing for many practical applications e.g. in machine learning.

Originally introduced for smallest enclosing ball problem and clustering by [6], the idea of a coreset is the following: instead of solving the original problem, one tries to identify a very small subset (coreset) of the points, such that the solution just on the coreset is guaranteed to be a good approximation of the true solution to the original problem. For the problem of finding the smallest enclosing ball of n points $P \in \mathbb{R}^d$, and $\epsilon > 0$, an ϵ-coreset S is a small subset of the points P such that the smallest enclosing ball of just S, blown up by a factor of $1+\epsilon$, contains all the original points P. It was shown that here ϵ-coresets of size $\lceil 1/\epsilon\rceil$ do always exist [4, 5] and that this is best possible [5]. This is very remarkable because the size of the coreset is independent of the dimension d of the space, and also independent of the number of points n, making it very attractive for the use in large scale problems (high n) and *kernel methods* (high d). This nice property is in contrast to many other geometric problems for which coresets usually have size exponential in the dimension, e.g. $\Theta(1/\epsilon^{(d-1)/2})$ for the extent problem [1]. For a nice review on existing coreset algorithms we refer to [2].

Just recently Clarkson [9] significantly widened the class of problems where the coreset idea can be applied, and showed that the nice property of constant $O(1/\epsilon)$ sized coresets indeed holds for the general problem of minimizing a convex function over the unit simplex.

Our Contributions and Related Work.

Following [9], we translate the coreset framework to the polytope distance problem of one polytope (w.r.t. the origin), distance between two polytopes, and hard- as well as soft-margin support vector machines, and introduce the geometric meaning of coresets and strong primal-dual approximation in this context.

We prove a new lower bound of $\left\lceil \frac{E^*}{\epsilon} \right\rceil + 1$ for the size of ϵ-coresets for polytope distance, where E^* is what we call the *excentricity* of the polytope. Together with the upper

[*] Research supported by the Swiss National Science Foundation (SNF Project 20PA21-121957)

bound of $\left\lceil \frac{E^*(1+o(1))}{\epsilon} \right\rceil$ as $\epsilon \to 0$, this shows that the size of obtained ϵ-coresets is asymptotically best possible. We also show tight bounds (up to a factor of two) for the distance problem between two polytopes.

For Gilbert's algorithm [13], one of the earliest known approximation algorithms for polytope distance, we give the first two proofs of convergence speed: First by observing that is in fact just an instance of the Frank Wolfe approximation algorithm for quadratic programs [11], which is now often called *sparse greedy approximation*, and secondly by giving a slightly easier geometric interpretation of a recent algebraic proof of [9]. Also, we generalize Gilbert's algorithm to the distance problem between two polytopes, where we are able to prove the same convergence speed. Furthermore, we can get rid of the expensive search for a starting point in this case which in previous approaches needed time quadratic in the number of points [24, 31].

Applications to Machine Learning.

On the application side, it is our mission to apply concepts and algorithms from computational geometry to machine learning. *Support Vector Machines* (SVM) [8] are among the most established and successful classification tools in machine learning, where from the name it is not immediately clear that the concept refers to nothing else than the separation of two classes of points by a hyperplane, by the largest possible margin. From the formulation as a quadratic program it follows that the problem is equivalent to the polytope distance problem, either for one or for two polytopes, depending on which SVM variant is considered (See Section 5). The *Perceptron* [25] refers to the case where we search for any hyperplane that separates two point classes, not necessarily one of maximum margin. The term *kernel methods* summarizes SVMs and Perceptrons where the points are assumed to live in an implicit high-dimensional feature space where we just know their pairwise scalar-products which is then called the *kernel*.

Sparsity of solutions. Our main contribution is to relate the coreset concept to *sparsity* of solutions of kernel methods: Using our bounds for the size of coresets, we derive a new fundamental property of SVMs and Perceptrons, giving nearly matching upper and lower bounds on the sparsity of their solutions, a parameter which is absolutely crucial for the practical performance of these methods on large scale problems. More precisely we show that any solution for a SVM or Perceptron, attaining at least a fraction μ of the optimal margin, must have at least $\left\lceil \frac{E^*}{1-\mu} \right\rceil + 1$ many (or $\left\lceil \frac{\frac{1}{2}E^*}{1-\mu} \right\rceil + 2$ in the two class case) non-zero coefficients in the worst case, and that a solution with $\left\lceil \frac{E^*(1+o(1))}{1-\mu} \right\rceil$ many non-zero coefficients can always be obtained for all instances. We are not aware of any existing lower bounds on the sparsity in the literature.

Training SVM in linear time. For any fixed fraction $0 \leq \mu < 1$, we show that Gilbert's algorithm in time $O(n)$ finds a solution attaining at least a μ-fraction of the optimal margin to the SVM and Perceptron (no matter if a kernel is used or not). This contrasts most of the existing SVM training algorithms which run in time usually cubic in n, or then often have no theoretical approximation guarantees except from converging in a finite number of steps [23], or have guarantees only on the primal or dual objective value, but not both. Tsang et al. have already applied the smallest enclosing ball coreset approach to train SVMs under the name *Core Vector Machine* (CVM) [30, 28], for one particular SVM variant (ℓ_2-loss with regularized offset), in the case that all points have the same norm. In this case the smallest enclosing ball problem is equivalent to finding the distance of one polytope from the origin. In another work [15] directly used coresets to solve the problem of separating two polytopes by a hyperplane passing through the origin, but this is again equivalent to a one polytope distance problem. Both approaches are therefore generalized by [9] and this work, proving faster algorithm convergence and smaller coresets. Here we generalize the coreset methods further to the two polytope case, encompassing all the currently most used hard- and soft-margin SVM variants with arbitrary kernels, with both ℓ_1 and ℓ_2-loss, in particular the special case of the CVM [30, 29, 28] and [15], while obtaining faster convergence and smaller coresets. Our generalization shows that all of the mostly used SVM variants can be trained in time linear in the number of sample points n, i.e. using linearly many kernel evaluations, for arbitrary kernels. Up till now this was only known for the CVM case and for linear SVMs without using a kernel.

2. CONCEPTS AND DEFINITIONS

2.1 Polytope Distance

Let $P \subset \mathbb{R}^d$ be a finite set of points. We want to compute the shortest distance ρ of any point inside the polytope $conv(P)$ to the origin.

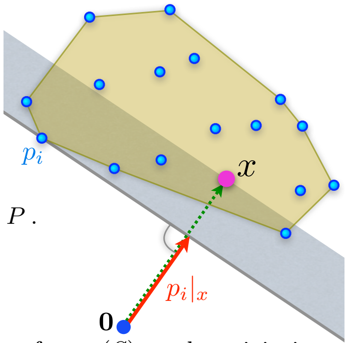
For $v, x \in \mathbb{R}^d$, Let $v|_x := \frac{\langle v,x \rangle}{\|x\|}$ denote the signed length of the *projection* of v onto the direction of the vector x.

Definition 1.
For any $\epsilon > 0$,

i) $x \in conv(P)$ is called an ϵ-*approximation*[1] to the optimal polytope distance, iff

$$\|x\| - p|_x \leq \epsilon \|x\| \quad \forall p \in P \, .$$

ii) A set of points $S \subseteq P$ with the property that the (optimal) closest point of $conv(S)$ to the origin is an ϵ-approximation to the distance of $conv(P)$ is called an ϵ-*coreset* of P.



Being an ϵ-approximation can thus be interpreted as the multiplicative gap between the "primal" distance $\|x\|$, and the "dual value" $\min_{p \in P} p|_x$, being small[2]. It is important

[1] Note that this is a *multiplicative* or relative approximation measure, where sometimes in the literature also *additive* ϵ-approximations are used. The corresponding coresets are sometimes called multiplicative ϵ-coresets to distinguish them from additive coresets [9].

[2] For minimizing general convex functions, the corresponding definition is that the gap between the primal and the dual value is small, which by weak duality also implies that the primal value is close to the optimum. A simple calculation shows that our geometric definition coincides with [9] for $f(x) := -\|Ax\|$, where $A \in \mathbb{R}^{d \times n}$ contains all points of P as columns.

to note that this definition of approximation is stronger than just requiring the distance x to be close to the optimal value:

LEMMA 1. *If x is an ϵ-approximation, then $(1-\epsilon)\|x\| \leq \rho \leq \|x\|$.*

PROOF. *RHS:* Clear by definition of the distance ρ. *LHS:* By definition of an ϵ-approximation, we have a closed halfspace (normal to x), with distance $(1-\epsilon)\|x\|$ from the origin, which contains $conv(P)$, which itself contains the optimal point x^* with $\|x^*\| = \rho$. □

Definition 2. We define the *excentricity* of a point set P as $E := \frac{D^2}{\rho^2}$, where $D := \max_{p,q \in P} \|p-q\|$ is the diameter of the polytope and ρ is the true polytope distance to the origin. Also, we define the *asymptotic excentricity* $E^* := \frac{R^2}{\rho^2}$, where we call $R := \max_{p \in P} \|p - c\|$ the radius of the polytope (where c is the unique point attaining the minimum distance to the origin).

It immediately follows that $E^* \leq E \leq 4E^*$ by triangle inequality ($R \leq D \leq 2R$). Also, the quantities E and E^* do correspond to the "non-linearity" defined by [9]: It holds that $C_f \leq \frac{\rho}{2}E$ and $C_f^* \leq \frac{\rho}{2}E^*(1+o(1))$ for $f(x) := -\|Ax\|$.

Definition 3. The *sparsity* of a convex combination $\sum_{i=1}^n \alpha_i p_i \in conv(P)$, $\sum_{i=1}^n \alpha_i = 1, \alpha_i \geq 0$ is the number of α_i that are non-zero.

By definition any ϵ-coreset of size s implies an ϵ-approximation of sparsity at most s.

2.2 Distance Between Two Polytopes

It is easy to see that the problem of finding the shortest distance between two polytopes is equivalent to finding the shortest vector in a single polytope, their Minkowski difference:

Definition 4. The *Minkowski difference*

$$MD(P_1, P_2) := \{u - v \mid u \in conv(P_1), v \in conv(P_2)\}$$

of two polytopes $conv(P_1)$ and $conv(P_2)$ is the set (in fact it is also a polytope [32]) consisting of all difference vectors.

Observe that $conv(P_1)$ and $conv(P_2)$ are separable by a hyperplane *iff* $\mathbf{0} \notin MD(P_1, P_2)$. We call a vector $x = x_1 - x_2$ an *ϵ-approximation* for the distance problem between the two polytopes $conv(P_1)$ and $conv(P_2)$ iff x is an ϵ-approximation for $MD(P_1, P_2)$. By the *sparsity* of a convex combination in $MD(P_1, P_2)$ we always mean the minimum number of non-zero coefficients of a representation as a difference of two convex combinations in the original polytopes. An *ϵ-coreset* is a subset $P_1' \cup P_2'$ of the two original point sets, $P_1' \subseteq P_1, P_2' \subseteq P_2$, such that the shortest vector in the restricted Minkowski difference $MD(P_1', P_2')$ is an ϵ-approximation.

Definition 5. We define the *excentricity* of a pair of two polytopes as $E_{P_1,P_2} := \frac{(D_1+D_2)^2}{\rho^2} = (\sqrt{E_1} + \sqrt{E_2})^2$ and the *asymptotic excentricity* as $E^*_{P_1,P_2} := \frac{(R_1+R_2)^2}{\rho^2} = (\sqrt{E_1^*} + \sqrt{E_2^*})^2$, with D_k, R_k denoting diameter and radius, ρ being the true distance between the two polytopes. For comparison, E_k, E_k^* are the "individual" excentricities of each polytope $conv(P_k)$, $k = 1, 2$.

This compares to the single polytope case as
$E_{P_1,P_2} \geq E_{MD(P_1,P_2)}$ and $E^*_{P_1,P_2} \geq E^*_{MD(P_1,P_2)}$.

3. LOWER BOUNDS ON THE SPARSITY OF ϵ-APPROXIMATIONS AND THE SIZE OF CORESETS

In this Section we will give two constructions of point sets, such that no small ϵ-coresets can possibly exist for the polytope distance problem. The geometric interpretation of these constructions is in fact very simple:

3.1 Distance of One Polytope from the Origin

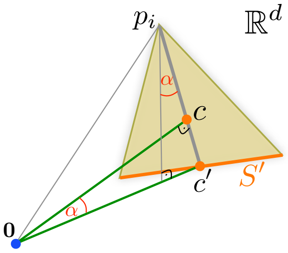
LEMMA 2. *For any given $0 < \epsilon < 1$, and for any $d \geq 2$, there exists a set of d points $P \subset \mathbb{R}^d$, such that*

i) any ϵ-coreset of P has size d (i.e. no strict subset can possibly be an ϵ-coreset).

ii) any ϵ-approximation of P has sparsity exactly d.

iii) any vector $x \in conv(P)$, satisfying $\frac{p|_x}{\rho} \geq 1-\epsilon \;\; \forall p \in P$, has sparsity exactly d.

iv) the excentricity of $conv(P)$ is $E = 2\epsilon d$ and the asymptotic excentricity is $E^* = \epsilon(d-1)$.



PROOF. By definition we already know that i) ⇐ ii) ⇐ iii), but we will prove the former statement first:

i) Let $\lambda > 0$ be a real parameter to be fixed later, and let our points be $p_j := \lambda e_j + (1-\lambda)c \in \mathbb{R}^d$ for $j \in [d]$, with the barycenter $c = (\frac{1}{d}, \ldots, \frac{1}{d})^T$ being the point closest to the origin of the standard $(d-1)$-simplex in $\mathbb{R}^d$. I.e. we just scale the unit simplex from its barycenter.

We will now show that for this particular set of points P, for some suitable choice of λ, no strict subset can possibly be an ϵ-coreset. To do so, let p_i be an arbitrary point of P and denote by c' the barycenter $c' := \frac{1}{d-1}\sum_{j \neq i} p_j$ which is the point closest to the origin of the sub-simplex $S' := conv(P \setminus \{p_i\})$.

By definition we have $\|c'-c\|^2 = \frac{\lambda^2}{d(d-1)}$, $\|c'-p_i\|^2 = \frac{\lambda^2 d}{d-1}$ and $\|c'\|^2 = \frac{d-1+\lambda^2}{d(d-1)}$. From planar geometry in the triangle c', p_i and the origin we have that $\sin(\alpha) := \frac{\|c'\| - p_i|_{c'}}{\|c'-p_i\|} = \frac{\|c'-c\|}{\|c'\|}$, which implies $1 - \frac{p_i|_{c'}}{\|c'\|} = \frac{\|c'-c\|\|c'-p_i\|}{\|c'\|^2} = \frac{\sqrt{\frac{\lambda^2}{d(d-1)}}\sqrt{\frac{d\lambda^2}{d-1}}}{\frac{d-1+\lambda^2}{d(d-1)}}$ $= \frac{d\lambda^2}{d-1+\lambda^2}$.

Now if we choose our parameter $\lambda := \sqrt{\epsilon}$, the above term on the right hand side is $\frac{d\epsilon}{d-1+\epsilon} > \epsilon$. In other words we now have that $\|c'\| - p_i|_{c'} > \epsilon\|c'\|$, so we have shown that no strict subset of the points can possibly be an ϵ-coreset.

ii) Using the construction above, we have shown that the barycenter $c' \in S'$ is not an ϵ-approximation, because it results in an insufficient approximation ratio $\frac{p_i|_{c'}}{\|c'\|} < 1 - \epsilon$. Now for every other point $x \in S'$, we can show that the ratio becomes even worse, if we argue as follows: In the denominator we know that c' is the point in S' of minimum norm, and in the numerator it holds that $p_i|_{c'} \geq p_i|_x \;\; \forall x \in S'$. The last inequality follows from the fact that the distance of a point x to a linear space is always at most as large as the distance to a subspace of it — or more formally if $p^{(x)}, p^{(c')}$ are the two projections of p_i onto x and c' respectively, we get that $\|p_i - p^{(x)}\| \geq \|p_i - p^{(c')}\|$ since $x \in \text{lin}(S')$ and $p_i|_{c'} = \|p^{(c')}\| = p_i|_{\text{lin}(S')}$. But by the Pythagorean theorem this implies $p_i|_x \leq p_i|_{c'}$. We have shown that no ϵ-approximation of sparsity $< d$ can possibly exist for our given point set.

iii) Let again $\lambda^2 := \epsilon$, and suppose $x \in S' = conv(P \backslash \{p_i\})$ for some $p_i \in P$ is such a convex combination of sparsity $\leq d-1$. We use the above result and calculate $\frac{p_i|_x}{\rho} \leq \frac{p_i|_{c'}}{\rho} = \|c'\|(1 - \frac{d\epsilon}{d-1+\epsilon})\sqrt{d} = \sqrt{\frac{d-1+\epsilon}{d(d-1)}} \frac{(d-1)(1-\epsilon)}{d-1+\epsilon} \sqrt{d} = \sqrt{\frac{d-1}{d-1+\epsilon}}(1 - \epsilon) < 1 - \epsilon$.

iv) It is straightforward to calculate that our point set has diameter $D^2 = \|p_1 - p_2\|^2 = 2\lambda^2$, radius $R^2 = \lambda^2 \frac{d-1}{d}$, and true distance $\rho^2 = \|c\|^2 = \frac{1}{d}$, so for the excentricity we obtain $E = 2\epsilon d$ and $E^* = \epsilon(d-1)$. □

THEOREM 3. *For any given $0 < \epsilon < 1$, for any $d \geq 2$, there exists a set of d points $P \subset \mathbb{R}^d$, such that the sparsity of any ϵ-approximation, and the size of any ϵ-coreset of P is at least*

$$\left\lceil \frac{\frac{1}{2}E}{\epsilon} \right\rceil \quad \textit{and} \quad \left\lceil \frac{E^*}{\epsilon} \right\rceil + 1$$

PROOF. The point set P from Lemma 2 satisfies $\frac{\frac{1}{2}E}{\epsilon} = d$, and $\frac{E^*}{\epsilon} = d - 1$. □

Note that the bound using E is by a factor of 2 better than if we would just have used the trivial bound $E \leq 4E^*$ together with the result for E^*.

3.2 Distance Between Two Polytopes

OBSERVATION 1. *If we have two point sets, one consisting of just one point and the other consisting of d points, and we consider the polytope distance problem between the corresponding two polytopes, the lower bound of Lemma 2 directly applies to the Minkowski difference, resulting in the lower bounds $\left\lceil \frac{\frac{1}{2}E}{\epsilon} \right\rceil + 1$ and $\left\lceil \frac{E^*}{\epsilon} \right\rceil + 2$ because we always need the single point of the second class in any linear combination. In this case the pair excentricities $E_{P_1,P_2}, E^*_{P_1,P_2}$ coincide with the single polytope excentricities E, E^*.*

However, we can even generalize the lower bound construction of Lemma 2 to the distance problem between two polytopes spanned by equally sized point classes:

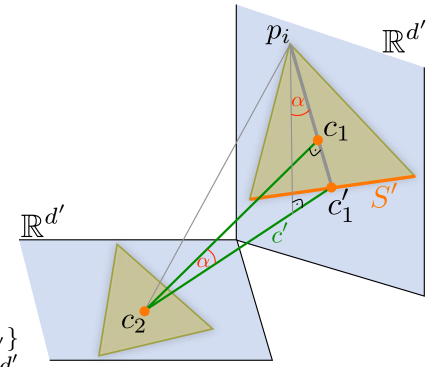
LEMMA 4. *For any given $0 < \epsilon < 1$, for any $d \geq 2$, there exist two equally sized point sets $P_1, P_2 \subset \mathbb{R}^{d=2d'}$, each consisting of d' points, such that*

i) any ϵ-coreset of $MD(P_1, P_2)$ has size d (i.e. no strict subset can possibly be an ϵ-coreset).

ii) any ϵ-approximation of $MD(P_1, P_2)$ has sparsity exactly d.

iii) any vector $x \in MD(P_1, P_2)$ satisfying $\frac{p|_x}{\rho} \geq 1-\epsilon \ \ \forall p \in MD(P_1, P_2)$ has sparsity d.

iv) the excentricity of the polytope pair is $E_{P_1,P_2} = 8\epsilon d'$ or $E^*_{P_1,P_2} = 4\epsilon(d' - 1)$ respectively.



PROOF. By definition we already know that i) $\Leftarrow$ ii) $\Leftarrow$ iii), but we will prove the former statement first:

Consider the two point classes $P_1 = \{p_1 \dots p_{d'}\}$ and $P_2 = \{p_{d'+1} \dots p_{2d'}\}$ living in $\mathbb{R}^d$, $d = 2d'$, with

$$p_i := \begin{cases} \lambda e_i + (1 - \lambda)c_1 & \text{for } 1 \leq i \leq d' \\ \lambda e_i + (1 - \lambda)c_2 & \text{for } d' + 1 \leq i \leq d \end{cases}$$

where $c_1 := (\frac{1}{d'}, \dots, \frac{1}{d'}, 0, \dots, 0)$ and $c_2 := (0, \dots, 0, \frac{1}{d'}, \dots, \frac{1}{d'})$. I.e. we have two 'copies' of scaled unit simplices. Obviously the shortest vector in the Minkowski difference is $c_1 - c_2$.

i) Analogously to the single-polytope case of Lemma 2, we will now show that for suitable λ, no strict subset of $P_1 \cup P_2$ is an ϵ-coreset of $MD(P_1, P_2)$. To do so, let p_i be an arbitrary point of P_1 and define $c' := c_1' - c_2$ where the barycenter $c_1' := \frac{1}{d'-1} \sum_{j \neq i} p_j$ is the point of the subsimplex $S' := conv(P_1 \backslash \{p_i\})$ closest to c_2. It is easy to check that c' is indeed the new shortest distance after removal of p_i.

By definition we have $\|c_1' - c_1\|^2 = \frac{\lambda^2}{d'(d'-1)}$, $\|c_1' - p_i\|^2 = \frac{\lambda^2 d'}{d'-1}$ and $\|c'\|^2 = \frac{2(d'-1)+\lambda^2}{d'(d'-1)}$. From planar geometry in the triangle c_1', p_i and c_2 we again have that $\sin(\alpha) := \frac{\|c'\| - (p_i - c_2)|_{c'}}{\|c_1' - p_i\|} = \frac{\|c_1' - c_1\|}{\|c'\|}$, which implies $1 - \frac{(p_i - c_2)|_{c'}}{\|c'\|} = \frac{\|c_1' - c_1\| \|c_1' - p_i\|}{\|c'\|^2} = \frac{\sqrt{\frac{\lambda^2}{d'(d'-1)}} \sqrt{\frac{d'\lambda^2}{d'-1}}}{\frac{2(d'-1)+\lambda^2}{d'(d'-1)}} = \frac{d'\lambda^2}{2(d'-1)+\lambda^2}$.

Now if we choose our parameter $\lambda := \sqrt{2\epsilon}$, the above term on the right hand side is $\frac{d' 2\epsilon}{2(d'-1)+2\epsilon} > \epsilon$. In other words we now have that $\|c'\| - (p_i - c_2)|_{c'} > \epsilon \|c'\|$, so we have shown that no strict subset of the $2d'$ points can possibly be an ϵ-coreset.

ii) The argument that the approximation ratio $\frac{p|_x}{\|x\|}$ is indeed best for the distance vector $x := c'$ — and thus there really is no ϵ-approximation of sparsity $< d$ — is the same as in the proof of Lemma 2, and additionally using that $(p_i - y_2)|_{c'} = (p_i - c_2)|_{c'} \ \ \forall y_2 \in conv(P_2)$, as c' is orthogonal on $conv(P_2)$.

iii) Let again $\lambda^2 := 2\epsilon$, and suppose $x \in S' = conv(P_1 \setminus \{p_i\})$ for some $p_i \in P_1$ is such a convex combination of sparsity $\leq d - 1$. We use the above result and calculate $\frac{(p_i - c_2)|_x}{\rho} \leq \frac{(p_i - c_2)|_{c'}}{\rho} = \|c'\|(1 - \frac{d' 2\epsilon}{2(d'-1)+2\epsilon})\sqrt{\frac{d'}{2}} = \sqrt{\frac{2(d'-1)+2\epsilon}{d'(d'-1)}} \frac{(d'-1)(1-\epsilon)}{d'-1+\epsilon} \sqrt{\frac{d'}{2}} = \sqrt{\frac{d'-1}{d'-1+\epsilon}}(1 - \epsilon) < 1 - \epsilon$.

iv) Check that each of our two polytopes has diameter $D^2 = 2\lambda^2$ and radius $R^2 = \lambda^2 \frac{d'-1}{d'}$. Since the optimal distance $\rho^2 = \frac{2}{d'}$, it follows that the pair excentricity is $E_{P_1,P_2} = \frac{(\sqrt{2}\lambda + \sqrt{2}\lambda)^2}{2/d'} = \frac{8\lambda^2}{2/d'} = 8\epsilon d'$ and the asymptotic pair excentricity is $E^*_{P_1,P_2} = \frac{\left(\sqrt{\frac{d'-1}{d'}}\lambda + \sqrt{\frac{d'-1}{d'}}\lambda\right)^2}{2/d'} = \frac{4 \frac{d'-1}{d'} \lambda^2}{2/d'} = 4\epsilon(d' - 1)$. □

THEOREM 5. *For any given $0 < \epsilon < 1$, for any $d \geq 2$, there exist two equally sized point sets $P_1, P_2 \subset \mathbb{R}^{d=2d'}$, each consisting of d' points, such that the sparsity of any ϵ-approximation of $MD(P_1, P_2)$, and the size of any ϵ-coreset is at least*

$$\left\lceil \frac{\frac{1}{4}E_{P_1,P_2}}{\epsilon} \right\rceil \quad \textit{and} \quad \left\lceil \frac{\frac{1}{2}E^*_{P_1,P_2}}{\epsilon} \right\rceil + 2\ .$$

PROOF. P_1 and P_2 from Lemma 4 satisfy $\frac{\frac{1}{4}E_{P_1,P_2}}{\epsilon} = 2d' = d$, and $\frac{\frac{1}{2}E^*_{P_1,P_2}}{\epsilon} + 2 = 2(d'-1) + 2 = d$. □

4. UPPER BOUNDS: ALGORITHMS TO CONSTRUCT CORESETS

4.1 Gilbert's Algorithm

The following gradient descent algorithm was introduced by Frank and Wolfe [11] as an approximation algorithm for quadratic programs. Since then it has independently been proposed again several times under different names; for polytope distance it is known as Gilbert's algorithm [13], where in the machine learning literature it is sometimes called *sparse greedy approximation*. The simplest general version of the algorithm is well summarized in [9, Algorithm 1.1], and provides ϵ-approximations of sparsity $O(\frac{1}{\epsilon})$ for any convex minimization problem on the standard simplex [9].

ALGORITHM 1. (GILBERT'S APPROXIMATION ALGORITHM FOR POLYTOPE DISTANCE [13]). *Start with $x_1 := p_0$, $p_0 \in P$ being the closest point to the origin. In step i, find the point $p_i \in P$ with smallest projection $p_i|_{x_i}$, and move to x_{i+1} being the point on the line segment $[x_i, p_i]$ which is closest to the origin. We stop as soon as x_{i+1} is an ϵ-approximation.*

Note that in order to run the algorithm, we only need to compute the projections of all points onto a given direction, and find the point closest to the origin on a line. Both can easily be achieved by evaluating scalar products, thus the algorithm works fine for *kernel methods*. Also, it can directly run on the Minkowski difference for the two polytope case.

Variants and applications. Gilbert's geometric algorithm has been applied to SVM training in the case of hard-margin as well as soft-margin with both ℓ_2-loss [16] and ℓ_1-loss [18, 19]. A variant of Gilbert's algorithm, the GJK algorithm, is a popular algorithm for collision detection in 3 dimensional space [12]. Another important variant of this, called the MDM algorithm [20], is in fact equivalent to one of the most used SVM training algorithms, SMO [23, 17]. For SVM training, [16] obtained good experimental results with a combination of Gilbert's and the MDM algorithm.

Convergence speed and running time. All mentioned algorithms in the above paragraph have in common that they converge, were successfully applied in practice, but no convergence speed or bound on the running time has ever been proved so far. Here we prove the convergence speed for Gilbert's algorithm, on one hand by observing for the first time that it is nothing else than the Frank-Wolfe algorithm [11] applied to the standard quadratic programming formulation of polytope distance[3], and on the other hand by giving a new slightly easier geometric variant of recent proofs by [3, 9] on the convergence speed of sparse greedy approximation:

For the following analysis, let $f_i := \|x_i\|$ be the current distance, $h_i := f_i - \rho$ be the primal error, and let $g_i := f_i - \omega_i$ denote the 'primal-dual' gap of our current estimate, with $\omega_i := \min_{p \in P} p|_{x_i}$. The key fact enabling linear convergence is the following bound, originally due to [3]:

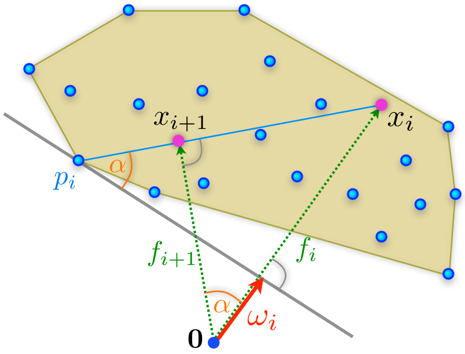


Figure 1: A step of Gilbert's algorithm.

LEMMA 6 (GEOMETRIC VARIANT OF [9, THEOREM 2.1]). *In each step of Gilbert's algorithm, the improvement in the primal error h_i is at least*

$$h_i - h_{i+1} \geq \frac{1}{2E\rho} g_i^2$$

PROOF. Suppose x_{i+1} is perpendicular to the line segment $[x_i, p_i]$: Then we have $f_i - f_{i+1} = (1 - \cos\alpha) f_i$. Using the inequality[4] $1 - \cos\alpha \geq \frac{1}{2}\sin^2\alpha$, we get

$$(1 - \cos\alpha) f_i \geq \frac{1}{2}\sin^2\alpha f_i = \frac{g_i^2}{2\|p_i - x_i\|^2} f_i \qquad (1)$$

but now we use the fact that since both x_i and p_i are inside $conv(P)$, $\|p_i - x_i\|$ is at most D. Now we already have proven the claim, since by definition $f_i \geq \rho$:

$$h_i - h_{i+1} = f_i - f_{i+1} \geq \frac{\rho}{2D^2} g_i^2 = \frac{1}{2E\rho} g_i^2 \qquad (2)$$

The case that x_{i+1} is at the endpoint p_i might in fact never occur: this would contradict the fact that the starting point for the algorithm was the point of shortest norm (since f_i decreases in each step). □

THEOREM 7. *Gilbert's algorithm succeeds after at most $2\left\lceil \frac{2E}{\epsilon} \right\rceil$ many steps.*

PROOF. Using Lemma 6, we can now follow along the same lines as in [9, Theorem 2.3]: If we switch to a re-scaled version of the error-parameters, $h_i' := \frac{1}{2E\rho} h_i$, $g_i' := \frac{1}{2E\rho} g_i$, then the inequality becomes

$$h_i' - h_{i+1}' \geq {g_i'}^2 \geq {h_i'}^2 \qquad (3)$$

($g_i \geq h_i$ does always hold by definition) or equivalently $h_{i+1}' \leq h_i'(1 - h_i')$: Plugging in $1 - \gamma \leq \frac{1}{1+\gamma}$ for $\gamma \geq -1$

[3] The quadratic programming formulation is $\min_x f(x) = (Ax)^2, x_i \geq 0, \sum_i x_i = 1$ when A is the $d \times n$-matrix containing all points as columns. Then the gradient $\nabla f(x)^T = A^T(Ax)$ consists of the scaled projections of all points onto the current vector Ax, so the Frank-Wolfe-Algorithms choice (see [9, Algorithm 1.1]) of the coordinate that minimizes the gradient is equivalent to moving towards the point with minimum projection, as in Gilbert's Algorithm [13].

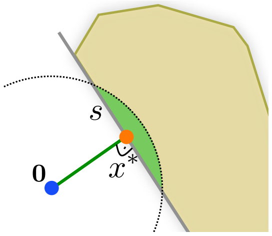
[4] This inequality is equivalent to $(1 - \cos\alpha)^2 \geq 0$.

gives $h'_{i+1} \leq \frac{h'_i}{1+h'_i} = \frac{1}{1+\frac{1}{h'_i}}$. Then by induction it is easy to obtain $h'_k \leq \frac{1}{k+1}$ and therefore $h'_k < \epsilon'$ for $k \geq K := \lceil \frac{1}{\epsilon'} \rceil$, if we can just show the induction hypothesis that $h'_1 \leq \frac{1}{2}$.

But this follows since our starting point is the point of shortest norm, which implies x_2 will always see x_1 and the origin in a right angle, therefore $\|x_1 - x_2\|^2 = f_1^2 - f_2^2 \leq D^2$, which implies $f_1 - f_2 \leq \frac{D^2}{2\rho}$ and therefore ${h'_1}^2 \leq {g'_1}^2 \leq h'_1 - h'_2 \leq \frac{1}{4}$ by using (3).

Now we have obtained small h'_i, but this does not necessarily imply yet that g'_i is also sufficiently small. For this we continue Gilbert's algorithm for another K steps, and suppose that in these subsequent steps, g' always remains $\geq \epsilon'$, then we always have that $h'_i - h'_{i+1} \geq {\epsilon'}^2$, and so $h'_K - h'_{2K} \geq K{\epsilon'}^2 \geq \epsilon'$, but this implies that $h'_{2K} < 0$, a contradiction. Thus for some $K \leq k \leq 2K$, we must have that $g'_k < \epsilon'$.

If we choose our $\epsilon' := \frac{1}{2E}\epsilon$, we know that after at most $2K = 2\lceil \frac{2E}{\epsilon} \rceil$ steps of the algorithm, the obtained primal-dual error is $g_k < \epsilon\rho \leq \epsilon f_k$, thus x_k is an ϵ-approximation. □



OBSERVATION 2. (ASYMPTOTIC CONVERGENCE OF GILBERT'S ALGORITHM). *Note that if we are are already very close to the true solution x^*, i.e. assume $f_i - \rho$ is small, say $h_i = f_i - \rho < \gamma$ for some $\gamma > 0$, then the inequality $\|p_i - x_i\| \leq D$ can be improved as follows: Observe that x_i is inside the optimal halfspace of distance ρ from the origin (as the entire polytope is), intersected with the ball of radius $\rho+\gamma$ around the origin. Let s be the furthest distance from x^* of any point in this intersection area. It is easy to see that $s = O(\sqrt{\gamma})$ is also small.*

By triangle inequality we have $\|p_i - x_i\| \leq \|p_i - x^*\| + s$, so we get the stronger inequality $\|p_i - x_i\|^2 \leq R^2 + O(\sqrt{\gamma})$ in the proof of Lemma 6. Therefore Gilbert's algorithm always succeeds in at most $2\left\lceil \frac{2E\rho}{\gamma} \right\rceil + 2\left\lceil \frac{2(E^*+O(\sqrt{\gamma}))}{\epsilon} \right\rceil = 2\left\lceil \frac{2E^*(1+o(1))}{\epsilon} \right\rceil$ many steps, as $\epsilon \to 0$.

Note that the above analysis also proves the existence of ϵ-coresets of size $2\lceil \frac{2E}{\epsilon} \rceil$ (and of size $2\left\lceil \frac{2E^*(1+o(1))}{\epsilon} \right\rceil$ in the asymptotic notation), because the same improvement bound applies to the "exact" combinatorial algorithm [9, Algorithm 4.1] that, when adding a new point to the set, computes the exact polytope distance of the new set.

4.2 An Improved Version of Gilbert's Algorithm for Two Polytopes

Coresets for the distance between two polytopes. All coreset methods for the single polytope case can directly be applied to the distance problem between two polytopes $conv(P^{(1)})$ and $conv(P^{(1)})$ by just running on the Minkowski difference $MD(P_1, P_2)$. This already makes the coreset approach available for all machine learning methods corresponding to a two polytope problem (as for example the standard SVM), see Section 5.

However, using the Minkowski difference has two major disadvantages: On one hand every vertex of $MD(P_1, P_2)$ always corresponds to two original vertices, one from each point class. Apart from potentially doubling the coreset size, this is a very unfortunate restriction if the shapes of the two point classes are very unbalanced, as e.g. in the *one-against-all* approach for multi-class classification, as it will create unnecessarily many non-zero coefficients in the smaller class. On the other hand, to run Gilbert's algorithm (or also the abstract version [9, Algorithm 1.1] or the reduced hull variant [18]) on $MD(P_1, P_2)$, we have to determine the starting point of shortest norm and therefore have to consider all pairs of original points. Although this starting configuration was used in practice (see e.g. the DirectSVM [24] and SimpleSVM [31] implementations), this should definitely be avoided for large sets of points. We overcome both difficulties as follows:

An Improved Algorithm for Two Polytopes. The following modified algorithm maintains a difference vector $x_i^{(1)} - x_i^{(2)}$ between the two polytopes, with $x_i^{(1)} \in conv(P^{(1)})$ and $x_i^{(2)} \in conv(P^{(2)})$. We again fix some notation first: Let $f_i := \|x_i^{(1)} - x_i^{(2)}\|$ be the current distance, $h_i := f_i - \rho$ be the primal error, and let $\omega_i := \min_{p \in P^{(1)}, q \in P^{(2)}} (p - q)|_{(x_i^{(1)} - x_i^{(2)})}$ be the 'dual' value. Then we can interpret $g_i^{(1)} := \max_{p \in P^{(1)}} (p - x_i^{(1)})|_{(x_i^{(2)} - x_i^{(1)})}$ and $g_i^{(2)} := \max_{p \in P^{(2)}} (p - x_i^{(2)})|_{(x_i^{(1)} - x_i^{(2)})}$ as being the two contributions to the 'primal-dual'-gap, so that $g_i := f_i - \omega_i = g_i^{(1)} + g_i^{(2)}$, see Figure 2.

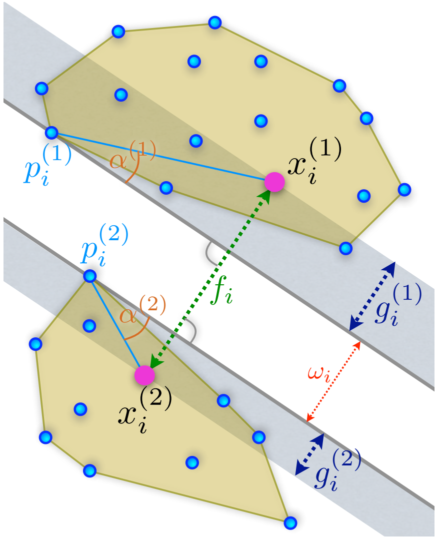


Figure 2: A step of Algorithm 2.

ALGORITHM 2. (IMPROVED GILBERT'S ALGORITHM FOR DISTANCE BETWEEN TWO POLYTOPES). *Start with an arbitrary point pair ($x_1^{(1)} \in P^{(1)}$, $x_1^{(2)} \in P^{(2)}$).*

In step i, $i \geq 1$, find the points $p_i^{(1)} \in P^{(1)}$ and $p_i^{(2)} \in P^{(2)}$ with smallest projection $(p_i^{(1)} - p_i^{(2)})|_{(x_i^{(1)} - x_i^{(2)})}$, and now decide in which of the two polytopes to do a Gilbert step: Choose the polytope $k \in \{1, 2\}$ for which the ratio $\frac{g_i^{(k)}}{\max\left\{\|x_i^{(k)} - p_i^{(k)}\|, \sqrt{g_i^{(k)} f_i}\right\}}$ is maximal, and move to $x_{i+1}^{(k)}$ being the point on the line segment $[x_i^{(k)}, p_i^{(k)}]$ which is closest to the "opposite" point $x_{i+1}^{(\bar{k})} := x_i^{(\bar{k})}$, which we keep unchanged. We stop as soon as $x_i^{(1)} - x_i^{(2)}$ is an ϵ-approximation.

Geometrically, the choice of k in the algorithm intuitively corresponds to choosing the polytope for which the angle $\alpha^{(k)}$ is largest (see Figure 2), and in the following we will show how this is beneficial for the improvement in each step.

Rectangular steps and hit steps. If the new point $x_{i+1}^{(k)}$ lies in the interior of the line segment $[x_i^{(k)}, p_i^{(k)}]$, we say that this step is a *rectangular step*, as indicated e.g. in the upper polytope in Figure 2. However if the new point ends up at the endpoint $p_i^{(k)}$ of the line segment, then we say that this is a *hit step* in polytope k, as e.g. in the lower polytope in Figure 2. It is not hard to see that a hit step in polytope k occurs if and only if $\|x_i^{(k)} - p_i^{(k)}\| \leq \sqrt{g_i^{(k)} f_i}$, and otherwise the step is rectangular. Note that in the single polytope case as in Lemma 6, hit steps are impossible by choice of the starting point, but here in the two polytope case this might indeed happen. From a computational perspective, hit steps are advantageous, as in each such step the number of points involved to describe the current approximation point inside one of the polytopes (called the number of *support vectors* in the SVM setting) decreases from a possibly large number down to one. However in the analysis of the algorithm, these hit steps pose some technical difficulties:

LEMMA 8. *The improvement in the primal error h_i in each step of Algorithm 2 is either*

$$h_i - h_{i+1} \geq \frac{1}{2\rho E_{P_1,P_2}} g_i^2 \tag{4}$$

or

$$h_i - h_{i+1} \geq C_{P_1,P_2}\, g_i \tag{5}$$

for $C_{P_1,P_2} := \frac{\rho}{4(\rho + D^{(1)} + D^{(2)})} \left(\min \frac{D^{(1)}}{D^{(2)}}, \frac{D^{(2)}}{D^{(1)}}\right)^2$ otherwise.

PROOF. $\boxed{\text{Case R}}$: In the case that the steps in both polytopes are rectangular we can follow the proof of Lemma 6: Assuming that the polytope chosen by the algorithm is k, we can follow (1) and (2) to get

$$\begin{aligned} h_i - h_{i+1} &\geq \tfrac{\rho}{2}\left(\frac{g_i^{(k)}}{\|x_i^{(k)} - p_i^{(k)}\|}\right)^2 \geq \tfrac{\rho}{2}\left(\max_{m=1,2} \frac{g_i^{(m)}}{D^{(m)}}\right)^2 \\ &\geq \tfrac{\rho}{2}\left(\frac{g_i^{(1)} + g_i^{(2)}}{D^{(1)} + D^{(2)}}\right)^2 = \tfrac{1}{2\rho E_{P_1,P_2}} g_i^2 \ , \end{aligned} \tag{6}$$

where we used $\max\left(\frac{a}{c}, \frac{b}{d}\right) \geq \frac{a+b}{c+d}$ $\forall a, b, c, d \geq 0$, Definition 5 of the Excentricity, and that $g_i = g_i^{(1)} + g_i^{(2)}$.

$\boxed{\text{Case M}}$: In the "mixed" case that the step in one polytope (r) is rectangular, but in the other polytope (h) is a hit step (due to $\|x_i^{(h)} - p_i^{(h)}\|^2 \leq g_i^{(h)} f_i$), we can argue as follows:

$\boxed{\text{Case M(r)}}$: If the algorithm chooses polytope (r), we can proceed analogously to (6) to obtain

$$h_i - h_{i+1} \geq \tfrac{\rho}{2} \frac{{g_i^{(r)}}^2}{\|x_i^{(r)} - p_i^{(r)}\|^2} = \tfrac{\rho}{2} \max\left\{\frac{{g_i^{(r)}}^2}{\|x_i^{(r)} - p_i^{(r)}\|^2}, \frac{{g_i^{(h)}}^2}{g_i^{(h)} f_i}\right\}, \tag{7}$$

but now there are two cases: If $g_i^{(h)} f_i \leq {D^{(h)}}^2$, then we use the same arithmetic trick as in (6),

$$h_i - h_{i+1} \geq \tfrac{\rho}{2}\left(\max_{m=1,2} \frac{g_i^{(m)}}{D^{(m)}}\right)^2 \geq \tfrac{1}{2\rho E_{P_1,P_2}} g_i^2. \tag{8}$$

However if $g_i^{(h)} f_i \geq {D^{(h)}}^2$, we have to argue differently: By the choice of the algorithm we know that $g_i^{(r)} \geq g_i^{(h)}$ because $\frac{g_i^{(r)}}{f_i} = \frac{{g_i^{(r)}}^2}{g_i^{(r)} f_i} \geq \frac{{g_i^{(r)}}^2}{\|x_i^{(r)} - p_i^{(r)}\|^2} \geq \frac{g_i^{(h)}}{f_i}$. As in (7), the improvement in the step can then be bounded by $h_i - h_{i+1} \geq \frac{\rho}{2} \frac{{g_i^{(r)}}^2}{\|x_i^{(r)} - p_i^{(r)}\|^2} \geq \frac{\rho}{2} \frac{{g_i^{(r)}}^2}{{D^{(r)}}^2} \geq \frac{\rho}{2} \frac{g_i^{(h)} g_i^{(r)}}{{D^{(r)}}^2}$ which by our assumption is $\geq \frac{\rho}{2} \frac{{D^{(h)}}^2}{{D^{(r)}}^2 f_i} g_i^{(r)} \geq \frac{\rho}{2} \frac{{D^{(h)}}^2}{{D^{(r)}}^2 f_i} \frac{1}{2}\left(g_i^{(r)} + g_i^{(h)}\right) \geq C_{P_1,P_2} g_i$. The last inequality follows because f_i is always smaller than $D^{(1)} + D^{(2)} + \rho$ by triangle inequality.

$\boxed{\text{Case M(h)}}$: If the choice criterium is $\frac{{g_i^{(r)}}^2}{\|x_i^{(r)} - p_i^{(r)}\|^2} \leq \frac{{g_i^{(h)}}^2}{g_i^{(h)} f_i}$, then algorithm will choose the polytope (h) where a hit step will occur. The improvement in a hit step is worst if $x_{i+1}^{(h)}$ ends up exactly at distance $\sqrt{f_i g_i^{(h)}}$ from $x_i^{(h)}$ — i.e. on the Thales ball over f_i — therefore $f_i^2 - f_{i+1}^2 \geq f_i g_i^{(h)}$ $\Rightarrow h_i - h_{i+1} = f_i - f_{i+1} \geq \frac{1}{2} g_i^{(h)} \geq \frac{\rho}{2} \frac{g_i^{(h)}}{f_i}$. Now if we again assume that $g_i^{(h)} f_i \leq {D^{(h)}}^2$, then analogously to (7), (8) we have $h_i - h_{i+1} \geq \frac{\rho}{2} \frac{g_i^{(h)}}{f_i} = \frac{\rho}{2} \max\left\{\frac{{g_i^{(r)}}^2}{\|x_i^{(r)} - p_i^{(r)}\|^2}, \frac{{g_i^{(h)}}^2}{g_i^{(h)} f_i}\right\} \geq \frac{1}{2\rho E_{P_1,P_2}} g_i^2$. On the other hand if $g_i^{(h)} f_i \geq {D^{(h)}}^2$, we distinguish two cases: First if $g_i^{(h)} \geq g_i^{(r)}$, then directly $h_i - h_{i+1} \geq \frac{1}{2} g_i^{(h)} \geq \frac{1}{4}(g_i^{(h)} + g_i^{(r)}) = \frac{1}{4} g_i \geq C_{P_1,P_2} g_i$. Secondly if $g_i^{(r)} \geq g_i^{(h)}$, we use that by the choice of the algorithm $\frac{g_i^{(h)}}{f_i} \geq \frac{{g_i^{(r)}}^2}{\|x_i^{(r)} - p_i^{(r)}\|^2} \geq \frac{{g_i^{(r)}}^2}{{D^{(r)}}^2}$, so we have $h_i - h_{i+1} \geq \frac{\rho}{2} \frac{{g_i^{(r)}}^2}{{D^{(r)}}^2} \geq C_{P_1,P_2} g_i$ analogous to the last part of the previous case M(r).

$\boxed{\text{Case H}}$: If there is a hit step in both polytopes, then we use that the algorithm has chosen the one with the larger value of $g_i^{(k)}$, therefore again by the "angle constraint" reasoning that for hit steps $f_i^2 - f_{i+1}^2 \geq f_i g_i^{(k)}$ we obtain $f_i - f_{i+1} \geq \frac{f_i}{2 f_i} g_i^{(k)} = \frac{1}{2} g_i^{(k)} \geq \frac{1}{4} g_i \geq C_{P_1,P_2} g_i$. □

THEOREM 9. *Algorithm 2 succeeds after at most $2\left\lceil \frac{2E_{P_1,P_2}}{\epsilon} \right\rceil + 3 + \frac{1}{C_{P_1,P_2}} \log \frac{D^{(1)} + D^{(2)}}{\rho\, C_{P_1,P_2}\, \epsilon} = O(1/\epsilon)$ many steps.*

PROOF. We count the steps of quadratic improvement (4) and those of linear improvement (5) separately, using that each kind of step results in strict improvement in the primal error h_i:

i) For quadratic improvement (4) we follow exactly along the proof of Theorem 7, for E_{P_1,P_2} being the pair excentricity: If we rescale by $h_i' := \frac{1}{2E_{P_1,P_2}\rho} h_i$, (4) gives $\frac{1}{h_{i+1}'} \geq 1 + \frac{1}{h_i'}$. Now we just use that initially h_1' is finite, therefore $\frac{1}{h_1'} \geq 0$ and by induction we get $\frac{1}{h_k'} \geq k - 1$ for all $k \geq 2$. It follows that $h_k' \leq \epsilon'$ for $k \geq K := \left\lceil \frac{1}{\epsilon'} \right\rceil + 1$. By the same argument as in the proof of Theorem 7, we have that after at most $2K = 2\left\lceil \frac{2E_{P_1,P_2}}{\epsilon} \right\rceil + 2$ many rectangular steps, $g_k < \epsilon\rho \leq \epsilon f_k$, thus $x_k^{(1)} - x_k^{(2)}$ is an ϵ-approximation.

ii) On the other hand we can bound the number of steps of linear improvement (5) by an easier argument: Let $C := C_{P_1,P_2}$. Now $h_i - h_{i+1} \geq C g_i \geq C h_i$ is equivalent to $h_{i+1} \leq (1 - C) h_i$ (recall that $0 < C < 1$). Using that initially $h_1 \leq D^{(1)} + D^{(2)}$, we get $h_k \leq (1 - C)^{k-1} h_1 \leq (1 - C)^{k-1}(D^{(1)} + D^{(2)})$ for all $k \geq 2$, which is $\leq \epsilon'$ as soon as $k - 1 \geq \frac{\log \frac{D^{(1)} + D^{(2)}}{\epsilon'}}{-\log(1 - C)}$, which in particular holds if $k - 1 \geq \frac{1}{C} \log \frac{D^{(1)} + D^{(2)}}{\epsilon'}$ by using the inequality $\lambda < -\log(1 - \lambda)$

for $0 < \lambda < 1$. For $\epsilon' := \rho C \epsilon$, this is enough because $\epsilon' \geq h_k \geq h_k - h_{k+1} \geq C g_k$ implies $g_k \leq \epsilon\rho \leq \epsilon f_k$, or in other words the algorithm obtains an ϵ-approximation after at most $\frac{1}{C} \log \frac{D^{(1)}+D^{(2)}}{\rho C \epsilon} + 1$ many steps of linear improvement. □

Generalization of our algorithm for convex optimization over products of simplices. We can also generalize our above new variant of sparse greedy approximation in terms of the general framework by Clarkson [9], when we extend it to solving any concave maximization problem over a product of finitely many simplices or convex hulls. To do so, we can prove the same step improvement (4) also for the case of convex functions defined on any product of simplices. We are currently investigating the details in this setting.

4.3 Smaller Coresets by "Away" Steps

Gilbert's algorithm and also its "exact" variant, due to their greedy nature, are not optimal in the size of the coresets they deliver. However, Clarkson showed that a modified procedure [9, Algorithm 5.1] based on an idea by [27], called the away steps algorithm, obtains smaller coresets. The following Theorem, together with our lower bounds from Section 3.1, will settle the question on the size of coresets for the distance problem of one polytope to the origin, because the size of the coreset obtained by the algorithm matches our lower bound, and therefore is best possible:

Theorem 10. *For any $\epsilon > 0$, the away steps algorithm [9, Algorithm 5.1] returns an ϵ-coreset of size at most $\left\lceil \frac{E}{\epsilon} \right\rceil$, and at most $\left\lceil \frac{E^*(1+o(1))}{\epsilon} \right\rceil$ as $\epsilon \to 0$.*

Proof. This follows directly from [9, Theorem 5.1], for $f(x) := -\|Ax\|$, when using the re-scaled $\epsilon' := \frac{\rho\epsilon}{2C_f}$, and applying $C_f \leq \frac{\rho}{2} E$ and $C_f^* \leq \frac{\rho}{2} E^*(1+o(1))$. □

Away steps in the case of two polytopes. We can adjust [9, Algorithm 5.1] for two polytopes as follows: Start with the *closest* point pair ($x_1^{(1)} \in P^{(1)}$, $x_1^{(2)} \in P^{(2)}$), and proceed as in [9, Algorithm 5.1]. In each step choose the polytope according to the choice criterium of our Algorithm 2, but with the modification that whenever a hit step is possible one either side, we choose to do this hit step.

Theorem 11. *For any $\epsilon > 0$, the modified away steps algorithm on two polytopes returns an ϵ-coreset of size at most $\left\lceil \frac{E_{P_1,P_2}}{\epsilon} \right\rceil$, and at most $\left\lceil \frac{E^*_{P_1,P_2}(1+o(1))}{\epsilon} \right\rceil$ as $\epsilon \to 0$.*

Proof. Sketch: We can follow the proof of [9, Theorem 5.1] using our Lemma 8, and observe that a hit step never increases the coreset size. The key point is that for any step that increases the coreset size, the improvement bound (4) always holds. The induction hypothesis $h_1' := \frac{1}{2E_{P_1,P_2}\rho} h_1 \leq \frac{1}{2}$ follows if we start at the closest pair. A more detailed proof will be made available in the full paper. □

5. APPLICATIONS TO MACHINE LEARNING

The advantage of the coreset approach is that both the running time of the algorithms and the sparsity of the obtained solutions is independent of the dimension d of the space and independent of the number of points n. This makes it very attractive for kernel methods, where the points are implicitly assumed to live in a (possibly very high dimensional) feature space.

Table 5 briefly recapitulates the fact that nearly all well known SVM variants are equivalent to a polytope distance problem between either one or two polytopes, showing that all these variants do fit into our framework of coresets. In the table, $x_i \in \mathbb{R}^d$, $1 \leq i \leq n$ denote the original points, $\phi(x_i)$ are their implicit images in the feature space defined by the kernel, and in the two class cases the labels of the points are given by $y_i = \pm 1$. ω and b are the normal and offset of the maximum margin hyperplane that we are searching for, and the ξ_i represent slack variables for the case of possible (punished) outliers.

5.1 Sparsity of SVM and Perceptron Solutions

The sparsity of kernel SVM and Perceptron solutions is *the* crucial ingredient for the performance of these methods on large scale problems: If we have an approximate solution ω, then still for every evaluation of the classifier (this means we are given a new "unseen" point and have to answer on which side of the hyperplane it lies), the scalar products to all points which appear with non-zero coefficient in ω (those are called the *support vectors*) have to be evaluated. The performance in practical use is therefore directly proportional to the sparsity of ω. Interestingly not much is known in the literature on this question, in particular no lower bounds are known to our knowledge. Using the above equivalences, we are now for the first time able to prove asymptotically matching upper and lower bounds for the sparsity of approximate SVM and Perceptron solutions:

Theorem 12. (Characterization of the Sparsity of Perceptron and SVM solutions using the Excentricity). *For any fraction $0 \leq \mu < 1$, the sparsity of an approximate solution attaining at least a μ-fraction of the optimal margin*[5]*, is bounded from above by $\left\lceil \frac{E}{1-\mu} \right\rceil$ and $\left\lceil \frac{E^*(1+o(1))}{1-\mu} \right\rceil$ as $\mu \to 1$ for the single polytope variants 1a),1b),1c) and by $\left\lceil \frac{E_{P_1,P_2}}{1-\mu} \right\rceil$ and $\left\lceil \frac{E^*_{P_1,P_2}(1+o(1))}{1-\mu} \right\rceil$ as $\mu \to 1$ for the two polytope variants 2a),2b) and 2c),2d)*[6]*.*

The sparsity is bounded from below by $\left\lceil \frac{\frac{1}{2}E}{1-\mu} \right\rceil$ and $\left\lceil \frac{E^*}{1-\mu} \right\rceil + 1$ for SVM variant 1a), and by $\left\lceil \frac{\frac{1}{4}E_{P_1,P_2}}{1-\mu} \right\rceil$ and $\left\lceil \frac{\frac{1}{2}E^*_{P_1,P_2}}{1-\mu} \right\rceil + 2$ for the standard SVM or Perceptron 2a).

Proof. *Upper bound:* This is a direct consequence of Theorem 10 in the single polytope case, and Theorem 11 in the two polytope case, showing that the away steps algorithm returns an $(1-\mu)$-coreset of the desired size, whose corresponding $(1-\mu)$-approximation proves our upper bound.

[5] In the single polytope case this means there is a separating hyperplane of distance $\mu\rho$ from the origin, whereas in the two polytope case it refers to a separating hyperplane such that all points have distance at least $\mu\frac{\rho}{2}$ from the plane.

[6] For the ℓ_1-loss SVM variants 2c),2d), our stated upper bound holds for the number of reduced hulls vertices [7, 10, 18] that are needed to represent a solution, however each vertex of a reduced hull corresponds to a fixed larger subset of non-zero coefficients when expressed in the original points. Thus the sparsity upper bound when expressed in the original points has to be multiplied by this factor, which for the ν-SVM variant 2d) is $\left\lceil \frac{\nu n}{2} \right\rceil$ [10].

	SVM Variant	**Primal Problem**	**Equivalent Polytope Distance Formulation**
1a	one-class SVM (hard-margin)	$\min_{w,\rho} \ \frac{1}{2}\|w\|^2 - \rho$ $w^T\phi(x_i) \geq \rho \ \ \forall i$	one polytope
1b	one-class ℓ_2-SVM (soft-margin)	$\min_{w,\rho,\xi} \ \frac{1}{2}\|w\|^2 - \rho + \frac{C}{2}\sum_i \xi_i^2$ $w^T\phi(x_i) \geq \rho - \xi_i \ \ \forall i$	one polytope [30, Equation (8)]
1c	two-class ℓ_2-SVM (with regularized or no offset)	$\min_{w,b,\rho,\xi} \ \frac{1}{2}(\|w\|^2 + b^2) - \rho + \frac{C}{2}\sum_i \xi_i^2$ $y_i(w^T\phi(x_i) - b) \geq \rho - \xi_i \ \ \forall i$	one polytope [30, Equation (13)], [15]
2a	two-class SVM, Perceptron (hard-margin)	$\min_{w,b} \ \frac{1}{2}\|w\|^2$ $y_i(w^T\phi(x_i) - b) \geq 1 \ \ \forall i$	two polytopes
2b	two-class ℓ_2-SVM (standard version)	$\min_{w,b,\xi} \ \frac{1}{2}\|w\|^2 + \frac{C}{2}\sum_i \xi_i^2$ $y_i(w^T\phi(x_i) - b) \geq 1 - \xi_i \ \ \forall i$	two polytopes [16, Section II]
2c	two-class ℓ_1-SVM (C-SVM)	$\min_{w,b,\xi} \ \frac{1}{2}\|w\|^2 + C\sum_i \xi_i$ $y_i(w^T\phi(x_i) - b) \geq 1 - \xi_i, \ \xi_i \geq 0 \ \ \forall i$	two (reduced) polytopes [7]
2d	two-class ℓ_1-SVM (ν-SVM)	$\min_{w,b,\rho,\xi} \ \frac{1}{2}\|w\|^2 - \rho + \frac{\nu}{2}\sum_i \xi_i$ $y_i(w^T\phi(x_i) - b) \geq \rho - \xi_i, \ \xi_i \geq 0 \ \ \forall i$	two (reduced) polytopes [10]

Table 1: SVM variants and their equivalent polytope distance formulations.

Lower bound: Any approximate solution that attains at least a μ-fraction of the optimal margin, is represented by a convex combination $x \in conv(P)$ (or $x \in MD(P_1, P_2)$ in the two polytope case) such that $p|_x \geq \mu\rho \ \ \forall p \in P$, or in other words $\frac{p|_x}{\rho} \geq 1 - \epsilon$ if we set $\epsilon := 1 - \mu$. By iii) of Lemma 2 (or Lemma 4 respectively), we have constructed a point set such that the sparsity of any such x has to be at least the claimed lower bound. □

Corollary 13. *The sparsity of any separating solution to a standard hard-margin two-class SVM or Perceptron is at least* $\lceil \frac{1}{2} E^*_{P_1,P_2} \rceil + 2$, *and at least* $\lceil \frac{1}{4} E_{P_1,P_2} \rceil$ *for some training point sets, whereas solutions of sparsity* $\lceil E_{P_1,P_2} \rceil$ *do always exist for all instances.*

Interpretation of the excentricity in the SVM and Perceptron case. For the Perceptron, [14] have proven a similar upper bound on the sparsity of separating solutions, and found it remarkable that it depends on the margin between the two classes. Our lower bound now confirms that this indeed has to be the case. For SVM, already [8, Section 7.5] conjectured, on the basis of empirical results, that it might be good to choose the free kernel parameters so that the quantity E is minimized. By our derived bounds we can now confirm that this choice is indeed good in the sense that it result in the best possible sparsity of the solutions. [8, Theorem 6] also showed that E gives an upper bound for the VC dimension of gap tolerant classifiers, a concept closely related to the complexity of the classification problem.

5.2 Linear Time Training of SVMs and Perceptrons

The following is a direct consequence of the analysis of Gilbert's Algorithm and our geometric interpretation of approximation in the one and two polytope setting:

Theorem 14. *For all SVM and Perceptron variants 1a) up to 2b), for arbitrary kernels, and for any fixed fraction* $0 \leq \mu < 1$, *we can find a solution attaining at least a* μ-*fraction of the optimal margin in time linear in the number of training points* n, *using Gilbert's Algorithm 1 or Algorithm 2 respectively.*

Proof. Theorem 7 and Theorem 9 show that the number of Gilbert steps needed is a constant independent of n and the dimension d. By keeping the lengths of all projections onto the previous estimate in memory, one can in each step update the all projections by just calculating n scalar products (of the new point p_i to all points in P) [16], therefore the number of kernel evaluations (scalar product computations) is n in each Gilbert step, and $O(n)$ in total. □

The above theorem also holds the reduced hull SVM variants 2c),2d), but there the number of kernel evaluations has to be multiplied with the previously mentioned factor[6].

Comparison to Existing SVM Training Algorithms. Our above result means that we removed the need for the detour of reducing SVM to a smallest enclosing ball problem, which was a theoretically and experimentally very successful method suggested by Tsang under the name *Core Vector Machine* (CVM) [30, 28], for the SVM variants 1b),1c), in the case that all points have the same norm. This is because in that special case the single polytope distance problem is equivalent to a smallest enclosing ball problem. The improved version of the CVM [28] uses Panigrahy's algorithm [22] to obtain a coreset of size $O(1/\epsilon^2)$ in the same number of steps. In another work [15] also proved the existence of coresets of size $O(1/\epsilon^2)$ for the problem of separating two polytopes by a hyperplane that goes through the origin, which is a special case of SVM variant 1c) and also equivalent to a single polytope distance problem.

Our contributions can be summarized as follows:

- By generalizing the coreset approach to the two polytope case, we encompass *all* the currently most used hard- and soft-margin SVM variants with arbitrary kernels, with both ℓ_1 and ℓ_2-loss.
- Our obtained coreset sizes — as well as the algorithm running times — are one order of magnitude smaller in terms of ϵ, and also have a smaller constant than most existing methods such as [30, 29, 28], [15].
- Our method works for arbitrary kernels, and is easier to apply in practice because we do not require the exact solution of small sub-problems, overcoming two disadvantages of [30, 29] and also [15].

Perceptrons. In the special case $\mu := 0$, our above Theorem gives a bound similar to the well known result that the traditional *Perceptron* algorithm [25] achieves perfect separation after $\frac{M^2}{\rho^2}$ many steps, where $M := \max_{p \in P} \|p\|$ is the largest norm of a sample point [21]. For cases of large margin, our bound of $2\lceil 2E\rceil$ steps is faster than the $\frac{M^2}{\rho^2}$ many steps guaranteed by the currently known bounds for (kernel) Perceptron algorithms [21, 26]. Another advantage of our result is that we can not only guarantee separation but simultaneously large margin.

6. REFERENCES

[1] P. K. Agarwal, S. Har-Peled, and K. R. Varadarajan. Approximating extent measures of points. *Journal of the ACM*, 51(4):606–635, 2004.

[2] P. K. Agarwal, S. Har-Peled, and K. R. Varadarajan. Geometric approximation via coresets. *Math. Sci. Res. Inst. Publ.*, 52:1–30, 2005.

[3] S. Ahipasaoglu, P. Sun, and M. Todd. Linear convergence of a modified Frank-Wolfe algorithm for computing minimum-volume enclosing ellipsoids. *Optimization Methods and Software*, 23(1):5–19, 2008.

[4] M. Bădoiu and K. L. Clarkson. Smaller core-sets for balls. *SODA '03: Proceedings of the fourteenth annual ACM-SIAM Symposium on Discrete Algorithms*, 2003.

[5] M. Bădoiu and K. L. Clarkson. Optimal core-sets for balls. *Computational Geometry: Theory and Applications*, 40(1):14–22, 2007.

[6] M. Bădoiu, S. Har-Peled, and P. Indyk. Approximate clustering via core-sets. *STOC '02: Proceedings of the thiry-fourth annual ACM Symposium on Theory of Computing*, 2002.

[7] K. Bennett and E. Bredensteiner. Duality and geometry in SVM classifiers. *ICML '00: Proceedings of the 17nd International Conference on Machine Learning*, 2000.

[8] C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.

[9] K. L. Clarkson. Coresets, sparse greedy approximation, and the Frank-Wolfe algorithm. *SODA '08: Proceedings of the nineteenth annual ACM-SIAM Symposium on Discrete Algorithms*, 2008.

[10] D. J. Crisp and C. J. C. Burges. A geometric interpretation of ν-SVM classifiers. *NIPS '00: Advances in Neural Information Processing Systems 12*, 2000.

[11] M. Frank and P. Wolfe. An algorithm for quadratic programming. *Naval Res. Logist. Quart.*, 3:95–110, 1956.

[12] E. Gilbert, D. Johnson, and S. Keerthi. A fast procedure for computing the distance between complex objects in three-dimensional space. *Robotics and Automation, IEEE Journal of*, 4(2):193–203, 1988.

[13] E. G. Gilbert. An iterative procedure for computing the minimum of a quadratic form on a convex set. *SIAM Journal on Control*, 4(1):61–80, 1966.

[14] T. Graepel, R. Herbrich, and R. C. Williamson. From margin to sparsity. *NIPS '00: Advances in Neural Information Processing Systems 12*, 2000.

[15] S. Har-Peled, D. Roth, and D. Zimak. Maximum margin coresets for active and noise tolerant learning. *IJCAI*, 2007.

[16] S. Keerthi, S. Shevade, C. Bhattacharyya, and K. Murthy. A fast iterative nearest point algorithm for support vector machine classifier design. *Neural Networks, IEEE Transactions on*, 11(1):124–136, 2000.

[17] J. López, Á. Barbero, and J. Dorronsoro. On the equivalence of the SMO and MDM algorithms for SVM training. *Machine Learning and Knowledge Discovery in Databases*, 288–300, 2008.

[18] M. E. Mavroforakis, M. Sdralis, and S. Theodoridis. A novel SVM geometric algorithm based on reduced convex hulls. *ICPR '06: 18th International Conference on Pattern Recognition*, 2:564–568, 2006.

[19] M. E. Mavroforakis and S. Theodoridis. A geometric approach to support vector machine (SVM) classification. *Neural Networks, IEEE Transactions on*, 17(3):671–682, 2006.

[20] B. Mitchell, V. Dem'yanov, and V. Malozemov. Finding the point of a polyhedron closest to the origin. *SIAM Journal on Control*, 1974.

[21] A. B. Novikoff. On convergence proofs for perceptrons. *Proceedings of the Symposium on the Mathematical Theory of Automata*, 12:615–622, 1963.

[22] R. Panigrahy. Minimum enclosing polytope in high dimensions. *CoRR*, cs.CG/0407020, 2004.

[23] J. C. Platt. Fast training of support vector machines using sequential minimal optimization. *Advances in kernel methods: support vector learning*, pages 185–208, 1999. MIT Press Cambridge, MA, USA.

[24] D. Roobaert. DirectSVM: A fast and simple support vector machine perceptron. *Neural Networks for Signal Processing X. Proceedings of the 2000 IEEE Signal Processing Society Workshop*, 1(1):356 – 365, 2000.

[25] F. Rosenblatt. The Perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.

[26] S. Shalev-Shwartz and Y. Singer. A new perspective on an old perceptron algorithm. *Learning Theory, Lecture Notes in Computer Science*, 264–278, 2005.

[27] M. Todd and E. Yildirim. On Khachiyan's algorithm for the computation of minimum-volume enclosing ellipsoids. *Discrete Applied Mathematics*, 155(13):1731–1744, 2007.

[28] I. W. Tsang, A. Kocsor, and J. T. Kwok. Simpler Core Vector Machines with enclosing balls. *ICML '07: Proceedings of the 24th International Conference on Machine Learning*, 2007.

[29] I. W. Tsang, J. T. Kwok, and J. Zurada. Generalized Core Vector Machines. *Neural Networks, IEEE Transactions on*, 17(5):1126–1140, 2006.

[30] I. W. Tsang, J. T. Kwok, and P.-M. Cheung. Core Vector Machines: Fast SVM training on very large data sets. *Journal of Machine Learning Research*, 6:363–392, 2005.

[31] S. V. N. Vishwanathan, A. J. Smola, and M. N. Murty. SimpleSVM. *ICML '03: Proceedings of the 20th International Conference on Machine Learning*, pages 760–767, 2003.

[32] G. M. Ziegler. Lectures on polytopes. *Graduate Texts in Mathematics*, 152, 1995. Springer Verlag.

APPENDIX

A. RELATION OF OUR GEOMETRIC CONCEPTS TO THE DEFINITIONS OF CLARKSON

Geometric v.s. algebraic primal-dual approximation. Here we show that our geometric definitions of ϵ-approximations and coresets do indeed fit to the general algebraic framework defined by [9]. To do so, we consider barycentric coordinates, and let A be the $d \times n$-matrix containing all n points of P as columns. Then the primal optimization problem of finding the point inside $conv(P)$ closest to the origin is

$$\min f(x) := \|Ax\| = \sqrt{x^T A^T Ax}, \text{ s.t. } \sum_{i=1}^{n} x_i = 1, x \geq 0,$$

and its Lagrange dual is

$$\max_{\alpha\in\mathbb{R},\beta_i\geq 0} \min_x \; L(x,\alpha,\beta) := f(x) - \alpha\left(\sum_{i=1}^{n} x_i - 1\right) - \sum_{i=1}^{n} \beta_i x_i .$$

Now since f is convex and differentiable (assume $\mathbf{0} \notin conv(P)$), this can equivalently be written as

$$\begin{array}{ll} \max\limits_{\alpha\in\mathbb{R},\beta_i\geq 0} & f(x) - (\alpha\mathbf{1}^T + (\beta_1,\dots,\beta_n))x + \alpha \\ \text{s.t.} & \nabla_x L(x,\alpha,\beta) = \mathbf{0} \end{array}$$

Computing the gradient $\nabla_x L(x,\alpha,\beta) = \frac{1}{\|Ax\|} x^T A^T A - \alpha\mathbf{1}^T - (\beta_1,\dots,\beta_n) = 0$ (observe that $\nabla_x f(x) = \frac{x^T A^T A}{\|Ax\|}$), we obtain the equivalent formulation

$$\begin{array}{ll} \max\limits_{x,\alpha\in\mathbb{R}} & f(x) - \dfrac{(Ax)^T}{\|Ax\|} Ax + \alpha = f(x) - f(x) + \alpha = \alpha \\ s.t. & \dfrac{(Ax)^T}{\|Ax\|} A - \alpha\mathbf{1}^T \geq 0 \end{array}$$

or equivalently

$$\begin{array}{c} \max_x \omega(x) \\ \text{with } \omega(x) := \min_{i\in[n]} \frac{(Ax)^T}{\|Ax\|} A_i \;, \end{array}$$

but this, as $\omega(x)$ is indeed the shortest projection of any point onto the current direction Ax, exactly matches our geometric definition of the dual, and our definition of ϵ-approximation is therefore equivalent to the multiplicative gap between primal and dual value being small. This shows that our geometric interpretation is indeed equivalent to Clarkson's algebraic approach, for this choice of f (up to swapping the sign of the objective function, as we do convex minimization, where the definitions of [9] are for concave maximization). Note that also for general convex optimization problems, weak duality shows that our approximation notion of "gap between primal and dual value being small" is stronger than just being close to the primal optimum.

The algebraic equivalent of the excentricity. Also, we can prove the immediate connection between our *excentricity* of a polyope to the *non-linearity* C_f as defined in [9, Section 2.2]. For $f(x) := \|Ax\|$, C_f will in fact directly correspond to $\frac{1}{2}E\rho$, the excentricity of the polytope $conv(P)$ scaled by the true distance ρ, as shown by the following Lemma:

LEMMA 15. *For $f(x) := \|Ax\|$, and D,R denoting diameter and radius of the polytope $conv(P)$, we have that*

$$C_f \leq \frac{D^2}{2\rho} = \frac{E\rho}{2} \;\; \textit{and} \;\; C_f^* \leq \frac{R^2(1+o(1))}{2\rho} = \frac{E^*\rho\;(1+o(1))}{2}$$

where $o(1)$ in the second inequality refers to $\epsilon \longrightarrow 0$.

PROOF. Let $S := \left\{x \in \mathbb{R}^n \;\middle|\; \sum_{i=1}^n x_i = 1,\; x_i \geq 0\right\}$ be the "unit simplex" spanned by the unit vectors in $\mathbb{R}^n$, so that we can write $conv(P) = AS$. Using the definition of C_f together with the Taylor expansion of f (see also [9, Section 2.2] in the journal version), and observing $\nabla_x^2 f(x) = \frac{A^T A}{\|Ax\|} - \frac{A^T Axx^T A^T A}{\|Ax\|^2}$, we can bound

$$\begin{array}{rcl} C_f & \leq & \sup\limits_{x,z,\tilde{x}\in S} \dfrac{1}{2}(z-x)^T \nabla_x^2 f(\tilde{x})(z-x) \\ & = & \sup\limits_{a,b,\tilde{b}\in AS} \dfrac{1}{2}\left(\dfrac{\|a-b\|^2}{\|\tilde{b}\|} - \dfrac{\|\tilde{b}^T(a-b)\|^2}{\|\tilde{b}\|^2}\right) \\ & \leq & \dfrac{diam(AS)^2}{2\rho} \end{array}$$

where in the last inequality we just omitted the righthand side term as it is always non-negative, and used the definition of the shortest distance ρ. The bound for C_f^* follows analogously, when restricting the supremum to the case where $b = Ax$ is already close to the optimum point (compare to Observation 2). □

Proof of Theorem 10 for the away step algorithm.

THEOREM 16. *For any $\epsilon > 0$, the away steps algorithm [9, Algorithm 5.1], when we run for a rescaled $\epsilon' := \frac{\rho\epsilon}{2C_f}$, returns an ϵ-coreset of size $\lceil \frac{E}{\epsilon} \rceil$, which is $\left\lceil \frac{E^*(1+o(1))}{\epsilon} \right\rceil$ as $\epsilon \longrightarrow 0$.*

PROOF. Using the above algebraic description of the polytope distance problem by $f(x) = \|Ax\|$, [9, Theorem 5.1] shows that the away steps algorithm delivers a subset of coordinate indices N of size $\lceil \frac{1}{\epsilon'} \rceil$ such that $f(x_N) - \omega(x_N) \leq 2C_f\epsilon'$, where x_N is the true optimal solution of the problem restricted to the coordinates in N. By the scaling $\epsilon' := \frac{\rho\epsilon}{2C_f}$, this means that in our setting, the away steps algorithm returns an ϵ-coreset of size $\left\lceil \frac{2C_f}{\rho\epsilon} \right\rceil \leq \lceil \frac{E}{\epsilon} \rceil$, and $\left\lceil \frac{2C_f^*(1+o(1)}{\rho\epsilon} \right\rceil \leq \left\lceil \frac{E^*(1+o(1)}{\epsilon} \right\rceil$ in the asymptotic case; the last two inequalities holding by Lemma 15. □