Ellipsoid Method Pres Notes

James Smith, Christopher Brissette, Benjamin Kushigian

May 2, 2019

1 Introduction

Let $P = \{x \in \mathbb{R}^n : Cx \leq d\}$ be a polytope, where C and d are integral.

Question. Is P empty?

This turns out to be a tricky problem, and for a long time it was not known to be solvable in polynomial time. There are a number of variants of the ellipsoid method, the most notable variant due to Leonid Khachiyan, who used it to solve a linear program over rational data in polytime. Here we present a variant that solves the strong nonemptiness problem on P. The **strong nonemptiness problem** asks, given P, whether or not P is empty, and if it is not, to produce a point witnessing P's nonemptiness.

The basic idea is simple enough: start with an initial ellipsoid E_0 guaranteed to contain P (or, more precisely, all of P's vertices), find the center point a_0 of E_0, which is explicit in E_0's representation, and test it for membership in P by checking against all the inequalities in $Cx \leq d$. If all of the inequalities are satisfied then we return a_0 as a witness of nonemptiness. Otherwise, we use our information to produce a new smaller ellipsoid E_1 and starting the process over again.

This process is repeated until we either witness nonemptiness or we reach until we can be confident that P is empty; termination is discussed in Section .

The following shows the basic algorithm:

Simple Algorithm

```
$E_0 =$ Some ball that contains $P$
while Not Terminated
    $E_{k+1} \leftarrow f(E_k)$
    if $a_k \in P$
        return Not Empty, $a_k$
    $k \leftarrow k + 1$
return Empty
```

This code is just going to be an overview used for the rest of the presentation, and will be delved into deeper as we move through the presentation.

2 Ellipsoids

2.1 Definitions

We should start by answering the question: **"what is an ellipsoid?"** To motivate this, let us recall the definition of an ellipse from calculus, namely the set of points satisfying

$$\left(\frac{x}{c}\right)^2 + \left(\frac{y}{d}\right)^2 \leq 1. \tag{2.1}$$

This form does not completely characterize all ellipses since it doesn't handle rotations and translations, but it's enough to get us started. We turn these constraints into matrix form:

$$\begin{pmatrix} x & y \end{pmatrix} \begin{pmatrix} c^{-2} & 0 \\ 0 & d^{-2} \end{pmatrix} \begin{pmatrix} x \\ t \end{pmatrix} \leq 1 \tag{2.2}$$

This matrix is positive definite, and we can generalize the definition of an ellipse to that of an ellipsoid:

Definition 1 (ellipsoid, $E(A, a)$)**.** For positive definite matrix $A \in \mathbb{R}^{n \times n}$, the set of points

$$E = E(A, a) = \{x \in \mathbb{R}^n : (x - a)^T A^{-1}(x - a) \leq 1\} \tag{2.3}$$

is called an **ellipsoid**.[a] Vector a is called the **center** of $E(A, a)$, and $E(A, a)$ is referred to as **the ellipsoid associated with** A **and** a. □

[a] Some sources define an ellipsoid to be the boundary ∂E by setting the inequality in (2.3) to be equality. For our purposes our current definition is more useful.

An ellipse is an ellipsoid

Example 1. The ellipse E in (2.1) may be put into matrix form in (2.2). Taking

$$A^{-1} = \begin{pmatrix} c^{-2} & 0 \\ 0 & d^{-2} \end{pmatrix}$$

we compute

$$A = \begin{pmatrix} c^2 & 0 \\ 0 & d^2 \end{pmatrix}$$

and we can write

$$E = E(A, 0).$$

2.2 Relating Ellipsoids and Spheres

The unit ball centered at a is a special case of the ellipsoid

$$\mathcal{B}(1, a) = E(I, a).$$

It turns out we can also think of the ellipsoid $E(A, a)$ as an affine transformation of $\mathcal{B}(1, 0)$, namely

$$E(A, a) = A^{1/2}\mathcal{B}(1, 0) + a. \tag{2.4}$$

Here $A^{1/2}$ is the unique positive definite matrix satisfying

$$A = A^{1/2} \cdot A^{1/2},$$

whose existence is a well-known property of positive definite matrices. The proof of (2.4) follows from some simple algebraic manipulation of the definition of an ellipsoid and is left to the reader as an exercise.

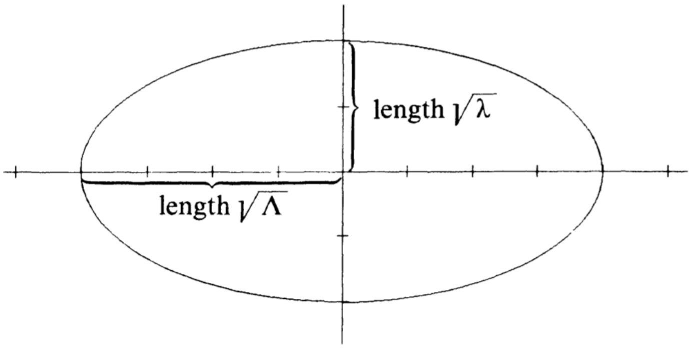

Figure 2.1: An ellipsoid

Example of an ellipsoid in R^2: connecting ellipsoids to eigenvalues

Example 2. Consider the ellipsoid in Figure 2.1. This ellipsoid may be written in the form $E(A, 0)$, where

$$A = \begin{pmatrix} \Lambda^2 & 0 \\ 0 & \lambda^2 \end{pmatrix},$$

where $\lambda < \Lambda$ are the two eigenvalues of A.

3 Initialization of the Ellipsoid Method

Now that we have a basic idea of how the ellipsoid method works, we would like to initialize the first ellipsoid E_0. We want to maintain the loop invariant that $P \in E_k \implies P \in E_{k+1}$, so E_0 should contain P. But P is specified by some arbitrary integral system of inequalities $Cx \leq d$, so how can we do this? Couldn't this correspond to an arbitrary polytope?

It turns out that we do have some bounds on P in relation to the encoding length of C and d, which is basically just our input size in bits.

Definition 2. The **encoding length** of an integer m is the number of bits needed to represent m, which can be calculated to be $1 + \lceil \log_2(|m|) \rceil$. The **encoding length** of integral vector v is the sum of the encoding lengths of the components of v, namely $\langle v \rangle = \sum_{i=1}^{n} \langle v \rangle_i$. Similarly, the **encoding length** of an integral matrix A is the sum of the encoding length of the k column vectors of C, namely $\langle C \rangle = \sum_{i=1}^{k} \langle c \rangle_i$.

We denote by $\langle C, d \rangle$ the encoding length of C and d, which is simply $\langle C \rangle + \langle d \rangle$. Next we provide a lemma that gives us an initial ellipsoid.

Lemma 1. $P \subseteq \mathcal{B}(R, 0)$, where

$$R := \sqrt{n} 2^{\langle C, d \rangle - n^2}. \tag{3.1}$$

Proof. LTR

□

4 Refining our Search

At this point we have our initial ellipsoid $E_0 = \mathcal{B}(R, 0)$ defined in the previous section, and we would like to fill out the body of the while loop. This entails transforming E_k to some other ellipsoid E_{k+1} along with a guarantee of some progress being made towards a final solution. In this section we show how to produce E_{k+1} from E_k, and in the following section we argue that this yields a terminating algorithm.

Suppose we have $E_k = E(A_k, a_k)$. The first question we ask is: *"is $a_k \in P$?"*. This can be answered by testing a_k against each linear inequality in $Cx \leq d$. If $c_i^t a_k \leq \gamma_i$ for each $i \in 1 \dots m$ then a_k is in P and we may return a_k as a witness of P's non-emptiness. Otherwise, during some comparison we found an inequality $c^t x \leq \gamma$ that was violated by a_k. We will use this to perform an ellipsoidal cut, and from there we will construct a new minimally-volumed ellipsoid containing the resulting convex body.

4.1 Ellipsoidal Cuts

Suppose we found inequality $c^T x \leq \gamma$ that is violated by center a_k. By convexity the hyperplane parallel to c through the center a, namely $\{x \in \mathbb{R}^n : c^T x = c^T a\}$,

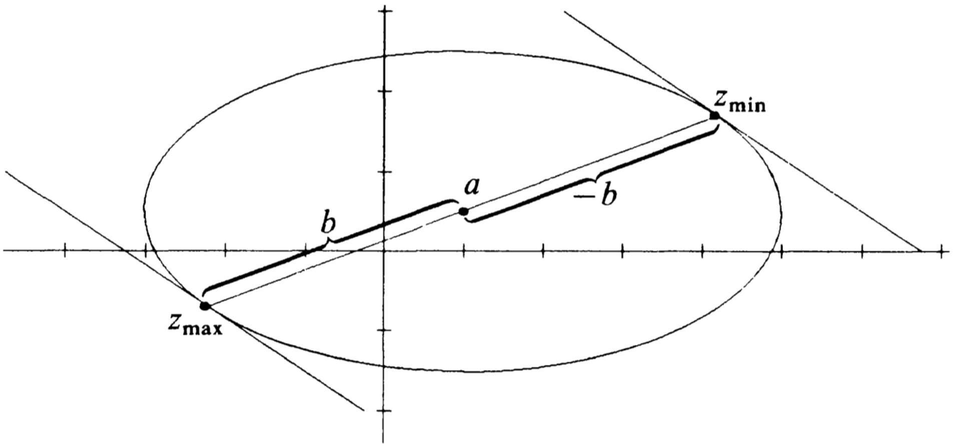

Figure 4.1: Optimizing a cost function c over an ellipsoid

does not intersect P, and by cutting E_k in half we have confined P to half of E_k:

$$E'(A_k, a_k, c) = E_k \cap \{x \in \mathbb{R}^n : c^T x \leq c^T a_k\}. \tag{4.1}$$

There are other cuts that we could have taken, but this one has the advantage of making other parts of our algorithm simpler.

We take a brief detour to derive a quantity that will be used later. We start with a question: how would one maximize a non-zero cost function c over ellipsoid $E(A, a)$? This is tricky, but we do know how to maximize over a unit ball $\mathcal{B}(1, a')$: the maximum value is found at vector $a' + \frac{c}{\|c\|}$. We define $Q := A^{1/2}$, and recall that $Q^{-1}E(A, a) = \mathcal{B}(1, Q^{-1}a)$ (which was left as a homework exercise). Then we compute as follows:

$$\begin{aligned}
\max\{c^T x | x \in E(A, a)\} &= \max\{c^T Q Q^{-1} x | Q^{-1} x \in Q^{-1} E(A, a)\} \\
&= \max\{c^T Q y | y \in \mathcal{B}(1, Q^{-1}a)\} \\
&= c^T Q \left(Q^{-1} a + \frac{Qc}{\|Qc\|} \right) \\
&= c^T \Big(a + \underbrace{\frac{Ac}{\sqrt{c^T A c}}}_{b} \Big)
\end{aligned}$$

Recalling that c^T is our cost function and a is the center of the ellipse we conclude that $b := Ac/\sqrt{c^T Ac}$ is the vector between a and the point on the boundary of E where c takes its maximal value; this is illustrated in Figure 4.1, where the tangent lines are parallel to the cost function c. Label z_{max} as the vector in E where c takes its maximal value, and label z_{min} as the vector in E where c takes its minimal value. It is clear that

$$z_{max} = a + b$$

$$z_{min} = a - b$$

This vector b turns out to be important for a later computation.

4.2 Löwner John Ellipsoids

Unfortunately, dividing ellipsoid E in half in (4.1) doesn't yield a new ellipsoid, so we don't yet have a new subproblem. Instead, we would like to find a new ellipsoid that is smaller than the first that contains the half ellipsoid E' (and therefore contains P). To this end we appeal to the following theorem, which is offered without proof.

Theorem 1. For every $K \subseteq \mathbb{R}^n$ there exists a unique ellipsoid E of minimal volume containing K.

Definition 3 (Löwner John Ellipsoid)**.** The **Löwner John** ellipsoid of a convex body K is the smallest ellipsoid that contains K, as given by Theorem 1.

After we've made our cut of E, resulting in E', we want to find the Löwner John ellipsoid of E'. In general the Löwner John ellipsoid of an arbitrary convex body k is very difficult to compute, but there are special cases that can be computed easily; in particular, for the half-ellipsoid $E'(A, a, c)$ from the previous section the following offers an explicit formula for computing the Löwner John ellipsoid.

$$\begin{aligned} b &= \frac{1}{\sqrt{c^T A_k c}} A_k c \\ a_{k+1} &= a_k - \frac{1}{n+1} b A_k c \\ A_{k+1} &= \frac{n^2}{n^2 - 1} \left(A_k - \frac{2}{n+1} b b^T \right) \end{aligned}$$

Two things to notice:

(a) a_{k+1} lies on the line from a to z_{min}

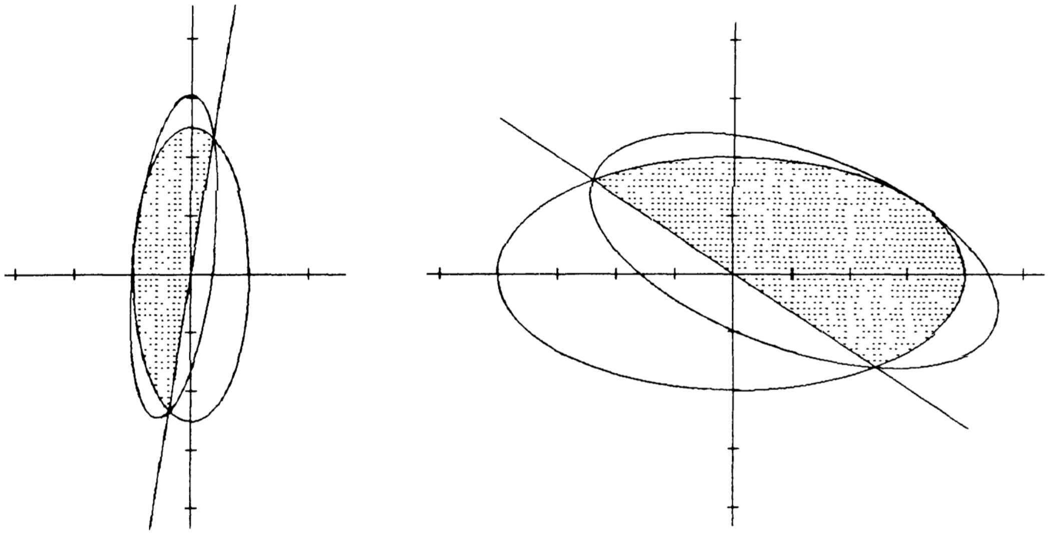

Figure 4.2: Two ellipsoids and their respective Löwner John ellipsoids corresponding to central cuts. Here the original ellipsoids are cenetered at the origin and squared with the xy-axes.

(b) $E(A_{k+1}, a_{k+1})$ touches $E(A, a)$ at z_{min} and the set

$$\partial E(A, a) \cap \{x | c^T x = c^T a\}$$

This intersection is an ellipsoidal projection into a lower dimension.

5 Termination

Finally, we would like to know if our algorithm terminates. If we ever find a point $a_k \in P$ then we return, so we are concerned with the case where a_k is never found to be in P. It turns out we can derive an upper bound N such that, if after considering $E(A_N, a_N)$ we still have not found an $a_k \in P$ we may conclude that P is empty; this follows from a volume argument. We offer the following two lemmas without proof.

Lemma 2.

$$\frac{\text{vol}(E_{k+1})}{\text{vol}(E_k)} = \left(\left(\frac{n}{n+1} \right)^{n+1} \left(\frac{n}{n-1} \right)^{n-1} \right)^{1/2} < e^{-1/2n} < 1$$

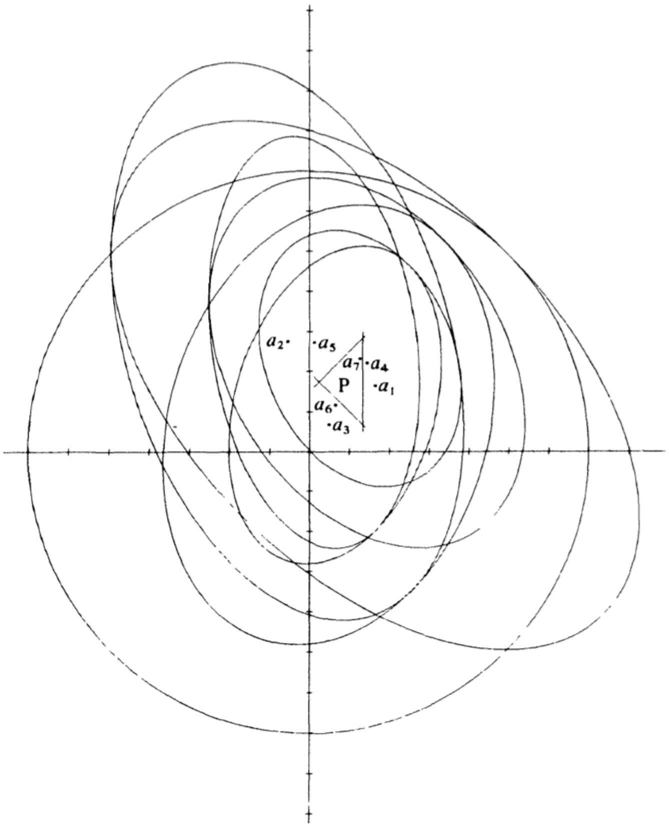

Figure 5.1: Several iterations of the ellipsoid method

Lemma 3. If P is a full-dimensional non-empty bounded polytope then

$$\text{vol}(P) \geq 2^{-(n+1)\langle C\rangle + n^3}$$

The above lemmas offer a shrinkage rate and a lower bound on P's volume. Since Lemma 2 guarantees that $\text{vol}(E_k) \to 0$ as $k \to \infty$, and since Lemma 3 offers us a positive lower bound on the volume, the loop invariant that E_k contains P guarantees that after some finite number of steps we will have found a point in P or that P is empty.

Finally, we'd like to use Lemma 2 to characterize how many iterations N will need to be run before we can conclude that $\text{vol}(E_N) < \text{vol}(P)$, which for non-empty P would violate $P \subseteq E_N$. We give, one last time, a lemma without proof.

Lemma 4. $\text{vol}(E_N) < \text{vol}(P)$, where $N := 2n((2n+1)\langle C\rangle + n\langle d\rangle - n^3)$.

Figure 5.1 illustrates several iterations of the ellipsoid method.

6 Algorithm and Polynomality

6.1 The Algorithm in Detail

Algorithm Pseudocode

```
1. Initialization:

       $k := 0$
       $N := 2n((2n+1)\langle C\rangle + n\langle d\rangle - n^3)$
       $A_0 := n2^{2(\langle C,d\rangle - n^2)}\mathbf{I}$
       $a_0 := 0$

2. General Step:

       IF $k = N$, NONE
       IF $a_k \in P$, STOP
       IF $a_k \notin P$, choose an inequality, $c^T x \leq \gamma$, of the system $Cx \leq d$
        that is violated by $a_k$

   Then set:

       $b := \frac{1}{\sqrt{c^T A_k c}} A_k c$
       $a_{k+1} := a_k - \frac{1}{n+1} b$
       $A_{k+1} := \frac{n^2}{n^2-1}(A_k - \frac{2}{n+1} bb^T)$
```

Algorithm Example

Example 3. Let us look at an example of our algorithm that terminates after a single iteration. Consider the system of inequalities below defining our polytope **P**:

$$\begin{pmatrix} 1 & 0 \\ -1 & 0 \\ 0 & 1 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \leq \begin{pmatrix} 1 \\ \frac{1}{2} \\ 1 \\ 0 \end{pmatrix} \tag{6.1}$$

This defines a rectangle in the positive quadrant in the $\mathbb{R}^2$ plane.

1. First we initialize:
   a) We choose $A_0 = 4\mathbf{I}$ to be our initial transform. Note the sphere this defines will have our polytope contained in it.
   b) $a_0 = 0$
2. Now we go into the general step:
   a) Obviously $a_0 \notin P$ and it violates the inequality:

$$\begin{pmatrix} -1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \leq -1 \tag{6.2}$$

   b) So now we solve for **b**:

$$c^T A_0 c = \begin{pmatrix} -1 & 0 \end{pmatrix} \begin{pmatrix} 4 & 0 \\ 0 & 4 \end{pmatrix} \begin{pmatrix} -1 \\ 0 \end{pmatrix} = \begin{pmatrix} 4 & 0 \end{pmatrix} \begin{pmatrix} -1 \\ 0 \end{pmatrix} = 4$$
$$\Rightarrow b := (c^T A_0 c)^{\frac{1}{2}} A_0 c = \begin{pmatrix} -2 \\ 0 \end{pmatrix}$$

   c) Now we solve for a_1

$$a_1 := a_0 - \frac{1}{3} \begin{pmatrix} -2 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{2}{3} \\ 0 \end{pmatrix}$$

   d) Finally we have to solve for A_1

$$A_1 := \frac{4}{3} \left(\begin{pmatrix} 4 & 0 \\ 0 & 4 \end{pmatrix} - \frac{2}{3} \begin{pmatrix} 4 & 0 \\ 0 & 0 \end{pmatrix} \right) = \begin{pmatrix} \frac{16}{9} & 0 \\ 0 & \frac{16}{3} \end{pmatrix}$$

   e) We now see that the algorithm terminates because $\begin{pmatrix} \frac{2}{3} & 0 \end{pmatrix}^T$ can be seen to be in our polytope.

6.2 Run time complexity

We can show that the initialization step requires $\mathcal{O}(mn)$ steps. Looking to the A_0 initialization:

$$A_0 := n2^{2(\langle C,d\rangle - n^2)}\mathbf{I}$$

$$\Rightarrow 2(\langle C, d\rangle - n^2) \text{ steps are required to calculate this.}$$

$$= 2(\langle C\rangle + \langle d\rangle - n^2)$$

$$= 2(\sum_{i,j}\langle c_{ij}\rangle + \sum_{j}\langle d_j\rangle) - n^2$$

$$= 2(\gamma mn + \lambda m - n^2)$$

$$m \geq n \Rightarrow mn \geq n^2$$

$$\therefore \text{ calculating } A_0 \text{ has time complexity } \mathcal{O}(mn)$$

Since each other initialization step is in constant time the entire initialization has run time complexity $\mathcal{O}(mn)$.

In the general step we can prove that substituting a_k into $Cx \leq d$ again requires $\mathcal{O}(mn)$ basic operations. Defining a_{k+1} and A_{k+1} require $\mathcal{O}(n)$ and $\mathcal{O}(n^2)$ elementary operations respectively. Since the algorithm completes in $N \propto \mathcal{O}(n^2\langle C, d\rangle)$ operations we therefore are left with $\mathcal{O}(mn^3\langle C, d\rangle)$ as the runtime complexity of the ellipsoid method.

6.3 Issues with Polynomality

As we can see the ellipsoid method gives us a theoretical time complexty that is polynomial! In practice this is often confounded by calculation however. First, notice that in general $a_k + 1$ and b are irrational, yielding small errors in our exact computation.

Say for instance that $\tilde{a_k}$ is a rounded version of the actual center a_k. Then the associated ellipsoid $\tilde{E_k} := E(A_k, \tilde{a_k})$ (which is the translated version of $E_k := E(A_k, a_k)$ containing the half-ellipsoid $E'_k = (A_k, a_k, c)$) may be such that $E'_{k-1} \not\subseteq \tilde{E_k}$. Note, because of this $P \subseteq E'_{k-1}$ may not be contained in $\tilde{E_k}$ nullifying the entire proof and method.

6.4 Adjusting for Error

For our rounding we use chopping to keep **p** digits of our representation. Due to the rounding error induced by this we need to adjust parameters to ensure **P** is contained inside our ellipsoid.

The way that we do this is by "blowing-up" our ellipsoid to contain **P**. We call our "blow-up" factor ε. Then we can show if:

$$\begin{aligned} N &:= 50(n+1)^2\langle C,d\rangle \\ p &:= 8N \\ \varepsilon &:= 1+\frac{1}{4(n+1)^2} \end{aligned}$$

then our proof method still works and terminates in polynomial time. However, as we can see this requires a problem-specific numerical precision **p**. This is obviously not ideal as computers tend to have set precision.

7 Homework Questions

7.1 Easy Problem

Recall that for positive definite $n \times n$ matrix A and n-vector a, the ellipsoid $E(A,a)$ is defined to be the set

$$E(A,a) = \{x \in \mathbb{R}^n : (x-a)^T A^{-1}(x-a)\} \leq 1.$$

Prove that the following is an equivalent definition,

$$E(A,a) = A^{1/2}\mathcal{B}(1,0) + a$$

where $\mathcal{B}(1,0)$ is the unit ball around the origin and$A^{1/2}$ is the unique positive definite matrix satisfying $A^{1/2}A^{1/2} = A$. *(hint: recall that positive definite matrices are symmetric).*

7.2 Medium Problem

Recall that the matrix A in $E(A,a)$ is symmetric positive definite. We have stated that because of this we can write $A = A^{\frac{1}{2}}A^{\frac{1}{2}}$. Prove that the symmetric matrix A can be written as $A = A^{\frac{1}{2}}A^{\frac{1}{2}}$ for some matrix $A^{\frac{1}{2}}$ if and only if A is positive semidefinite.

7.3 Hard Problem

3. In class we stated that the vertices of polytope P, where P is defined by

$$P = \{x \in \mathbb{R}^n : Cx \leq d\},$$

are always contained within a ball centered at 0 with radius

$$R := \sqrt{n} 2^{\langle C,d \rangle - n^2},$$

where $\langle C, d \rangle$ is the encoding length of C, d.

Cramer's rule tells us that for a non-singular matrix B and linear system $Bx \leq b$ with precisely one solution v, $v_i = \frac{\det(B_i)}{\det(B)}$, where v_i is the ith component of v and B_i is the matrix resulting from replacing the ith column of B with vector b.

Use Cramer's rule along with the fact that $|\det B_i| \leq 2^{\langle B_i \rangle - n^2} - 1$ to show that the vertices of P are always contained in the ball $B(R, 0)$.