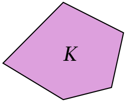
Ellipsoid Method

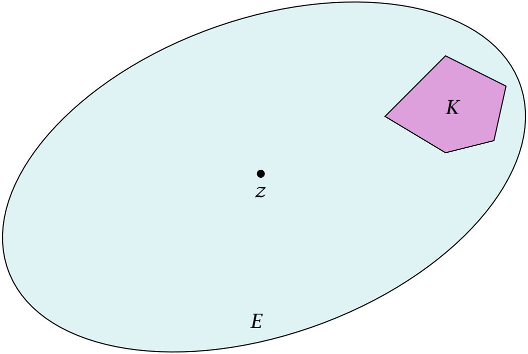
Ellipsoid Method

- Let K be a convex set.



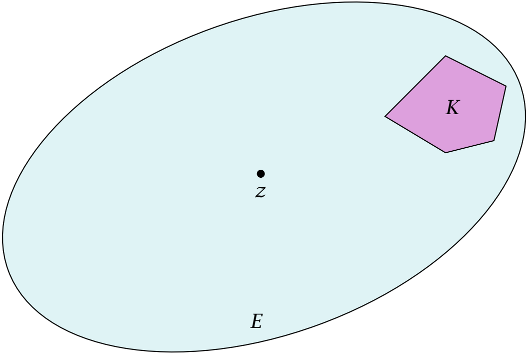
Ellipsoid Method

- Let K be a convex set.
- Maintain ellipsoid E that is guaranteed to contain K provided that K is non-empty.



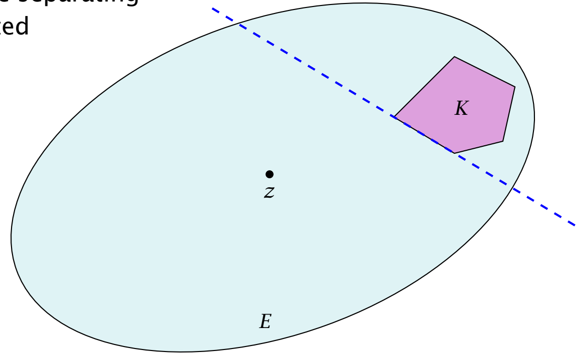
Ellipsoid Method

- Let K be a convex set.
- Maintain ellipsoid E that is guaranteed to contain K provided that K is non-empty.
- If center $z \in K$ STOP.



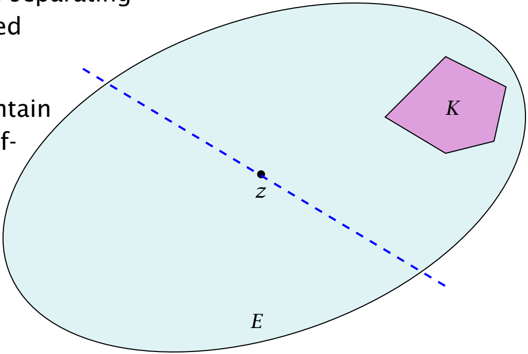
Ellipsoid Method

- Let K be a convex set.
- Maintain ellipsoid E that is guaranteed to contain K provided that K is non-empty.
- If center $z \in K$ STOP.
- Otw. find a hyperplane separating K from z (e.g. a violated constraint in the LP).



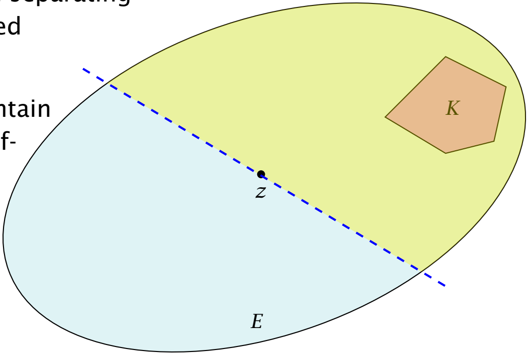
Ellipsoid Method

- Let K be a convex set.
- Maintain ellipsoid E that is guaranteed to contain K provided that K is non-empty.
- If center $z \in K$ STOP.
- Otw. find a hyperplane separating K from z (e.g. a violated constraint in the LP).
- Shift hyperplane to contain node z. H denotes half-space that contains K.



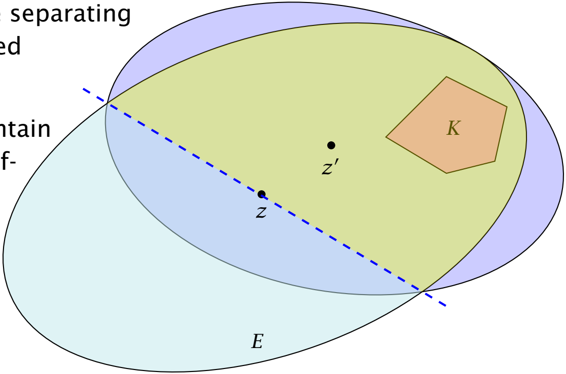
Ellipsoid Method

- Let K be a convex set.
- Maintain ellipsoid E that is guaranteed to contain K provided that K is non-empty.
- If center $z \in K$ STOP.
- Otw. find a hyperplane separating K from z (e.g. a violated constraint in the LP).
- Shift hyperplane to contain node z. H denotes half-space that contains K.



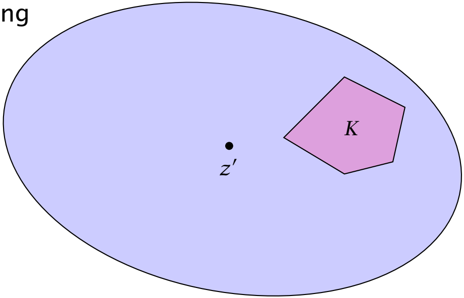
Ellipsoid Method

- Let K be a convex set.
- Maintain ellipsoid E that is guaranteed to contain K provided that K is non-empty.
- If center $z \in K$ STOP.
- Otw. find a hyperplane separating K from z (e.g. a violated constraint in the LP).
- Shift hyperplane to contain node z. H denotes half-space that contains K.
- Compute (smallest) ellipsoid E' that contains $K \cap H$.



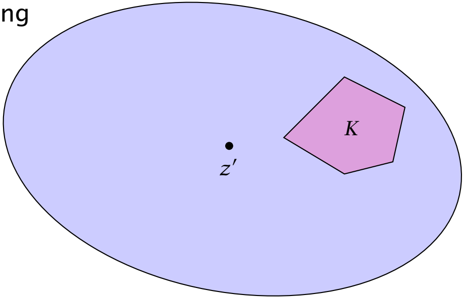
Ellipsoid Method

- Let K be a convex set.
- Maintain ellipsoid E that is guaranteed to contain K provided that K is non-empty.
- If center $z \in K$ STOP.
- Otw. find a hyperplane separating K from z (e.g. a violated constraint in the LP).
- Shift hyperplane to contain node z. H denotes half-space that contains K.
- Compute (smallest) ellipsoid E' that contains $K \cap H$.



Ellipsoid Method

- Let K be a convex set.
- Maintain ellipsoid E that is guaranteed to contain K provided that K is non-empty.
- If center $z \in K$ STOP.
- Otw. find a hyperplane separating K from z (e.g. a violated constraint in the LP).
- Shift hyperplane to contain node z. H denotes half-space that contains K.
- Compute (smallest) ellipsoid E' that contains $K \cap H$.
- REPEAT



Issues/Questions:

- How do you choose the first Ellipsoid? What is its volume?
- What if the polytop K is unbounded?
- How do you measure progress? By how much does the volume decrease in each iteration?
- When can you stop? What is the minimum volume of a non-empty polytop?

Definition 3

A mapping $f : \mathbb{R}^n \to \mathbb{R}^n$ with $f(x) = Lx + t$, where L is an invertible matrix is called an affine transformation.

Definition 4

A ball in $\mathbb{R}^n$ with center c and radius r is given by

$$\begin{aligned} B(c,r) &= \{x \mid (x-c)^t(x-c) \le r^2\} \\ &= \{x \mid \sum_i (x-c)_i^2/r^2 \le 1\} \end{aligned}$$

$B(0,1)$ is called the unit ball.

Definition 5

An affine transformation of the unit ball is called an ellipsoid.

Definition 5

An affine transformation of the unit ball is called an ellipsoid.

From $f(x) = Lx + t$ follows $x = L^{-1}(f(x) - t)$.

Definition 5

An affine transformation of the unit ball is called an ellipsoid.

From $f(x) = Lx + t$ follows $x = L^{-1}(f(x) - t)$.

$$f(B(0,1))$$

Definition 5

An affine transformation of the unit ball is called an ellipsoid.

From $f(x) = Lx + t$ follows $x = L^{-1}(f(x) - t)$.

$$f(B(0,1)) = \{f(x) \mid x \in B(0,1)\}$$

Definition 5

An affine transformation of the unit ball is called an ellipsoid.

From $f(x) = Lx + t$ follows $x = L^{-1}(f(x) - t)$.

$$\begin{aligned} f(B(0,1)) &= \{f(x) \mid x \in B(0,1)\} \\ &= \{y \in \mathbb{R}^n \mid L^{-1}(y - t) \in B(0,1)\} \end{aligned}$$

Definition 5

An affine transformation of the unit ball is called an ellipsoid.

From $f(x) = Lx + t$ follows $x = L^{-1}(f(x) - t)$.

$$\begin{aligned} f(B(0,1)) &= \{f(x) \mid x \in B(0,1)\} \\ &= \{y \in \mathbb{R}^n \mid L^{-1}(y - t) \in B(0,1)\} \\ &= \{y \in \mathbb{R}^n \mid (y - t)^t {L^{-1}}^t L^{-1}(y - t) \leq 1\} \end{aligned}$$

Definition 5

An affine transformation of the unit ball is called an ellipsoid.

From $f(x) = Lx + t$ follows $x = L^{-1}(f(x) - t)$.

$$\begin{aligned} f(B(0,1)) &= \{f(x) \mid x \in B(0,1)\} \\ &= \{y \in \mathbb{R}^n \mid L^{-1}(y-t) \in B(0,1)\} \\ &= \{y \in \mathbb{R}^n \mid (y-t)^t {L^{-1}}^t L^{-1}(y-t) \le 1\} \\ &= \{y \in \mathbb{R}^n \mid (y-t)^t Q^{-1}(y-t) \le 1\} \end{aligned}$$

Definition 5

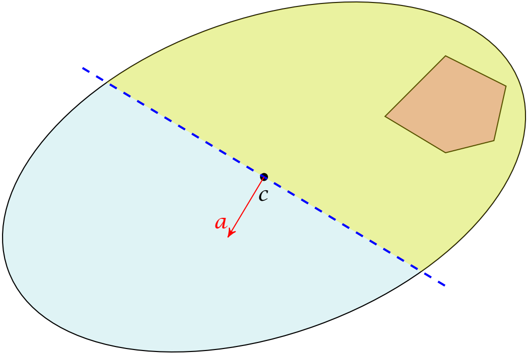
An affine transformation of the unit ball is called an ellipsoid.

From $f(x) = Lx + t$ follows $x = L^{-1}(f(x) - t)$.

$$\begin{aligned} f(B(0,1)) &= \{f(x) \mid x \in B(0,1)\} \\ &= \{y \in \mathbb{R}^n \mid L^{-1}(y-t) \in B(0,1)\} \\ &= \{y \in \mathbb{R}^n \mid (y-t)^t {L^{-1}}^t L^{-1}(y-t) \le 1\} \\ &= \{y \in \mathbb{R}^n \mid (y-t)^t Q^{-1}(y-t) \le 1\} \end{aligned}$$

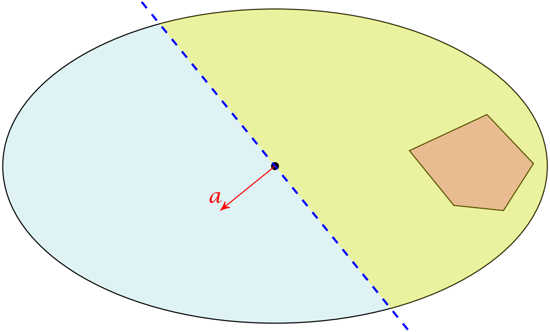
where $Q = LL^t$ is an invertible matrix.

How to Compute the New Ellipsoid



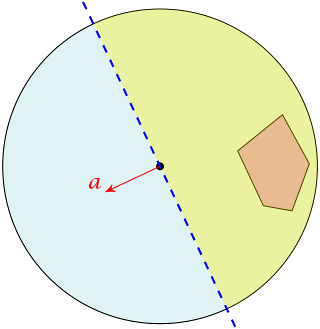
How to Compute the New Ellipsoid

- Use f^{-1} (recall that $f = Lx + t$ is the transformation function for the Ellipsoid) to rotate/distort the ellipsoid (back) into the unit ball.



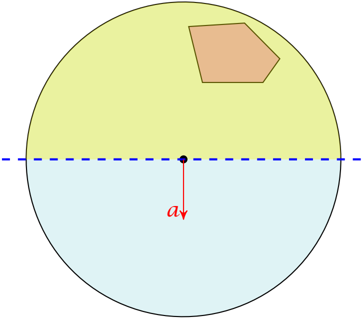
How to Compute the New Ellipsoid

- Use f^{-1} (recall that $f = Lx + t$ is the transformation function for the Ellipsoid) to rotate/distort the ellipsoid (back) into the unit ball.



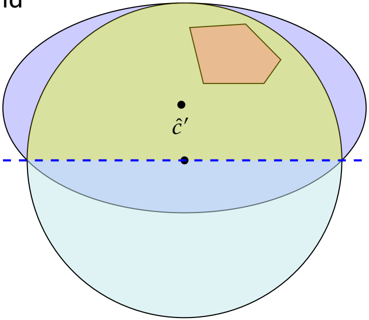
How to Compute the New Ellipsoid

- Use f^{-1} (recall that $f = Lx + t$ is the transformation function for the Ellipsoid) to rotate/distort the ellipsoid (back) into the unit ball.
- Use a rotation R^{-1} to rotate the unit ball such that the normal vector of the halfspace is parallel to e_1.



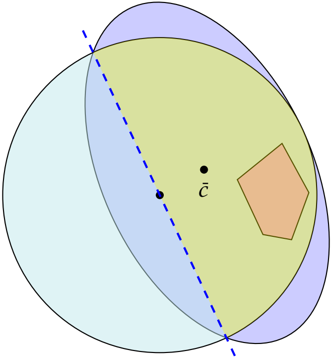
How to Compute the New Ellipsoid

- Use f^{-1} (recall that $f = Lx + t$ is the transformation function for the Ellipsoid) to rotate/distort the ellipsoid (back) into the unit ball.
- Use a rotation R^{-1} to rotate the unit ball such that the normal vector of the halfspace is parallel to e_1.
- Compute the new center $\hat{c}'$ and the new matrix $\hat{Q}'$ for this simplified setting.



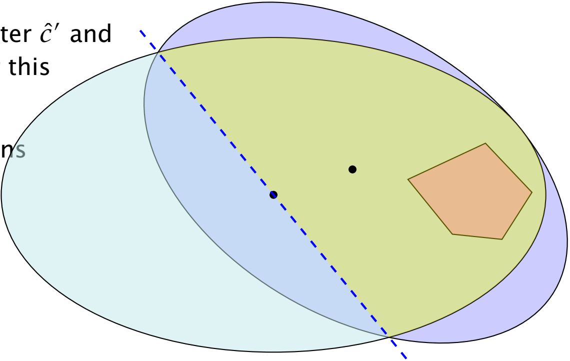
How to Compute the New Ellipsoid

- Use f^{-1} (recall that $f = Lx + t$ is the transformation function for the Ellipsoid) to rotate/distort the ellipsoid (back) into the unit ball.
- Use a rotation R^{-1} to rotate the unit ball such that the normal vector of the halfspace is parallel to e_1.
- Compute the new center $\hat{c}'$ and the new matrix $\hat{Q}'$ for this simplified setting.
- Use the transformations R and f to get the new center c' and the new matrix Q' for the original ellipsoid E.



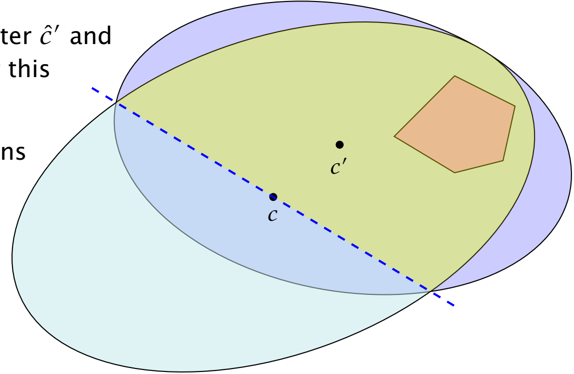
How to Compute the New Ellipsoid

- Use f^{-1} (recall that $f = Lx + t$ is the transformation function for the Ellipsoid) to rotate/distort the ellipsoid (back) into the unit ball.
- Use a rotation R^{-1} to rotate the unit ball such that the normal vector of the halfspace is parallel to e_1.
- Compute the new center $\hat{c}'$ and the new matrix $\hat{Q}'$ for this simplified setting.
- Use the transformations R and f to get the new center c' and the new matrix Q' for the original ellipsoid E.

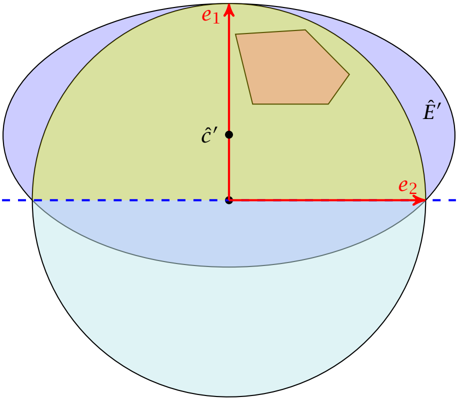


How to Compute the New Ellipsoid

- Use f^{-1} (recall that $f = Lx + t$ is the transformation function for the Ellipsoid) to rotate/distort the ellipsoid (back) into the unit ball.
- Use a rotation R^{-1} to rotate the unit ball such that the normal vector of the halfspace is parallel to e_1.
- Compute the new center $\hat{c}'$ and the new matrix $\hat{Q}'$ for this simplified setting.
- Use the transformations R and f to get the new center c' and the new matrix Q' for the original ellipsoid E.

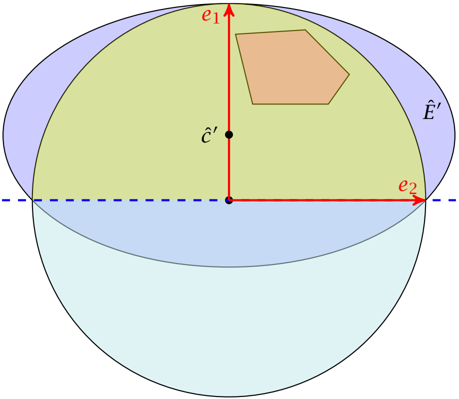


The Easy Case



- The new center lies on axis x_1. Hence, $\hat{c}' = te_1$ for $t > 0$.
- The vectors $e_1, e_2, \dots$ have to fulfill the ellipsoid constraint with equality. Hence $(e_i - \hat{c}')^t \hat{Q}'^{-1} (e_i - \hat{c}') = 1$.

The Easy Case



- The new center lies on axis x_1. Hence, $\hat{c}' = te_1$ for $t > 0$.
- The vectors $e_1, e_2, \dots$ have to fulfill the ellipsoid constraint with equality. Hence $(e_i - \hat{c}')^t {\hat{Q}'}^{-1}(e_i - \hat{c}') = 1$.

The Easy Case

- The obtain the matrix $\hat{Q}'^{-1}$ for our ellipsoid $\hat{E}'$ note that $\hat{E}'$ is axis-parallel.
- Let a denote the radius along the x_1-axis and let b denote the (common) radius for the other axes.
- The matrix

$$\hat{L}' = \begin{pmatrix} a & 0 & \dots & 0 \\ 0 & b & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & b \end{pmatrix}$$

  maps the unit ball (via function $\hat{f}'(x) = \hat{L}'x$) to an axis-parallel ellipsoid with radius a in direction x_1 and b in all other directions.

The Easy Case

- The obtain the matrix $\hat{Q}'^{-1}$ for our ellipsoid $\hat{E}'$ note that $\hat{E}'$ is axis-parallel.
- Let a denote the radius along the x_1-axis and let b denote the (common) radius for the other axes.
- The matrix

$$\hat{L}' = \begin{pmatrix} a & 0 & \dots & 0 \\ 0 & b & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & b \end{pmatrix}$$

maps the unit ball (via function $\hat{f}'(x) = \hat{L}'x$) to an axis-parallel ellipsoid with radius a in direction x_1 and b in all other directions.

The Easy Case

- The obtain the matrix $\hat{Q}'^{-1}$ for our ellipsoid $\hat{E}'$ note that $\hat{E}'$ is axis-parallel.
- Let a denote the radius along the x_1-axis and let b denote the (common) radius for the other axes.
- The matrix

  $$\hat{L}' = \begin{pmatrix} a & 0 & \dots & 0 \\ 0 & b & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & b \end{pmatrix}$$

  maps the unit ball (via function $\hat{f}'(x) = \hat{L}'x$) to an axis-parallel ellipsoid with radius a in direction x_1 and b in all other directions.

The Easy Case

- As $\hat{Q}' = \hat{L}'\hat{L}'^t$ the matrix $\hat{Q}'^{-1}$ is of the form

$$\hat{Q}'^{-1} = \begin{pmatrix} \frac{1}{a^2} & 0 & \dots & 0 \\ 0 & \frac{1}{b^2} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \frac{1}{b^2} \end{pmatrix}$$

The Easy Case

- $(e_1 - \hat{c}')^t \hat{Q}'^{-1}(e_1 - \hat{c}') = 1$ gives

$$\begin{pmatrix} 1-t \\ 0 \\ \vdots \\ 0 \end{pmatrix}^t \cdot \begin{pmatrix} \frac{1}{a^2} & 0 & \dots & 0 \\ 0 & \frac{1}{b^2} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \frac{1}{b^2} \end{pmatrix} \cdot \begin{pmatrix} 1-t \\ 0 \\ \vdots \\ 0 \end{pmatrix} = 1$$

- This gives $(1-t)^2 = a^2$.

The Easy Case

- For $i \neq 1$ the equation $(e_i - \hat{c}')^t \hat{Q}'^{-1}(e_i - \hat{c}') = 1$ gives

$$\begin{pmatrix} -t \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}^t \cdot \begin{pmatrix} \frac{1}{a^2} & 0 & \dots & 0 \\ 0 & \frac{1}{b^2} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \frac{1}{b^2} \end{pmatrix} \cdot \begin{pmatrix} -t \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} = 1$$

- This gives $\frac{t^2}{a^2} + \frac{1}{b^2} = 1$, and hence

$$\frac{1}{b^2} = 1 - \frac{t^2}{a^2}$$

The Easy Case

- For $i \neq 1$ the equation $(e_i - \hat{c}')^t \hat{Q}'^{-1}(e_i - \hat{c}') = 1$ gives

$$\begin{pmatrix} -t \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}^t \cdot \begin{pmatrix} \frac{1}{a^2} & 0 & \dots & 0 \\ 0 & \frac{1}{b^2} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \frac{1}{b^2} \end{pmatrix} \cdot \begin{pmatrix} -t \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} = 1$$

- This gives $\frac{t^2}{a^2} + \frac{1}{b^2} = 1$, and hence

$$\frac{1}{b^2} = 1 - \frac{t^2}{a^2} = 1 - \frac{t^2}{(1-t)^2}$$

The Easy Case

- For $i \neq 1$ the equation $(e_i - \hat{c}')^t \hat{Q}'^{-1}(e_i - \hat{c}') = 1$ gives

$$\begin{pmatrix} -t \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}^t \cdot \begin{pmatrix} \frac{1}{a^2} & 0 & \dots & 0 \\ 0 & \frac{1}{b^2} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \frac{1}{b^2} \end{pmatrix} \cdot \begin{pmatrix} -t \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} = 1$$

- This gives $\frac{t^2}{a^2} + \frac{1}{b^2} = 1$, and hence

$$\frac{1}{b^2} = 1 - \frac{t^2}{a^2} = 1 - \frac{t^2}{(1-t)^2} = \frac{1-2t}{(1-t)^2}$$

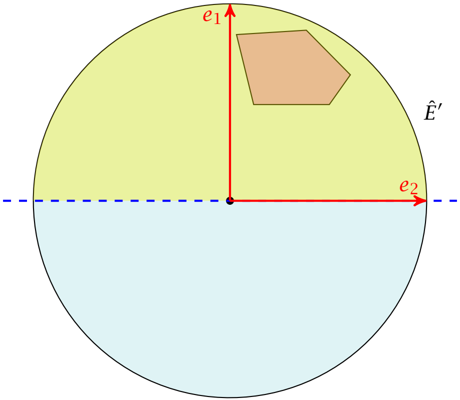
Summary

So far we have

$$a = 1 - t \quad \text{and} \quad b = \frac{1-t}{\sqrt{1-2t}}$$

The Easy Case

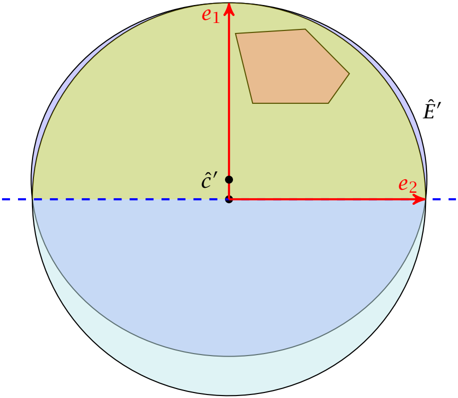
We still have many choices for t:



Choose t such that the volume of $\hat{E}'$ is minimal!!!

The Easy Case

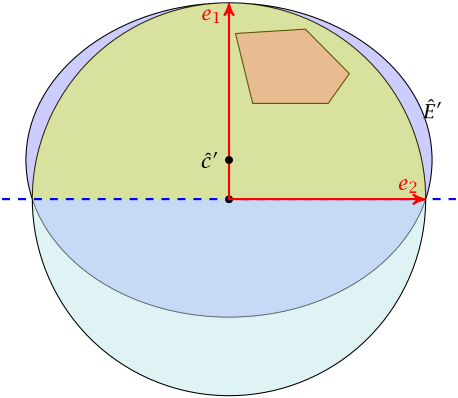
We still have many choices for t:



Choose t such that the volume of $\hat{E}'$ is minimal!!!

The Easy Case

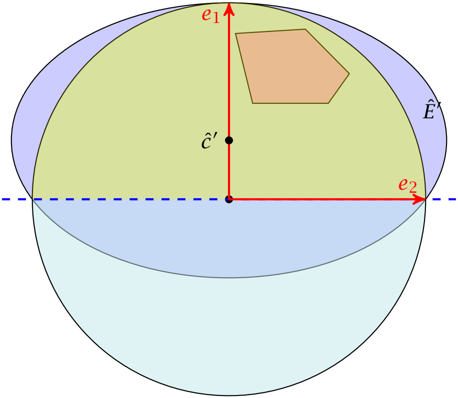
We still have many choices for t:



Choose t such that the volume of $\hat{E}'$ is minimal!!!

The Easy Case

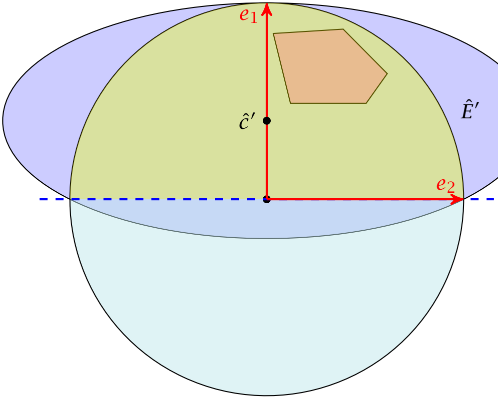
We still have many choices for t:



Choose t such that the volume of $\hat{E}'$ is minimal!!!

The Easy Case

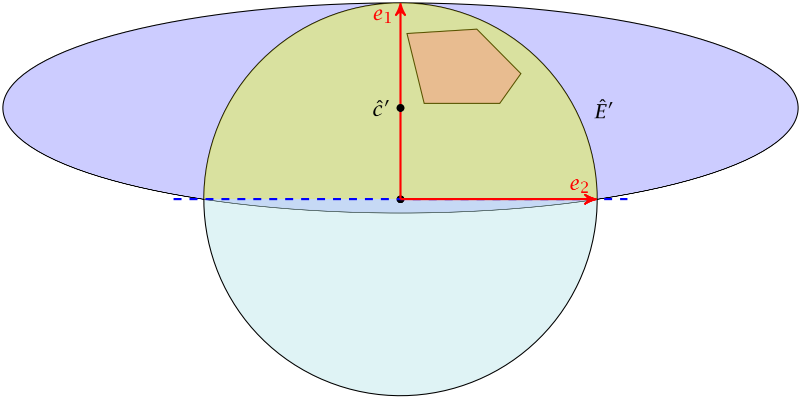
We still have many choices for t:



Choose t such that the volume of $\hat{E}'$ is minimal!!!

The Easy Case

We still have many choices for t:



Choose t such that the volume of $\hat{E}'$ is minimal!!!

The Easy Case

We want to choose t such that the volume of $\hat{E}'$ is minimal.

Lemma 6

Let L be an affine transformation and $K \subseteq \mathbb{R}^n$. Then

$$\mathrm{vol}(L(K)) = |\det(L)| \cdot \mathrm{vol}(K) \ .$$

The Easy Case

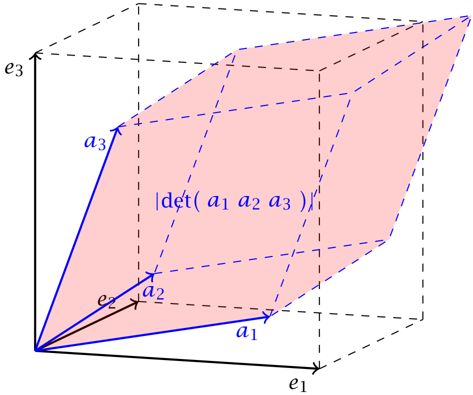
We want to choose t such that the volume of $\hat{E}'$ is minimal.

Lemma 6

Let L be an affine transformation and $K \subseteq \mathbb{R}^n$. Then

$$\mathrm{vol}(L(K)) = |\det(L)| \cdot \mathrm{vol}(K) \ .$$

n-dimensional volume



The Easy Case

- We want to choose t such that the volume of $\hat{E}'$ is minimal.

  $$\mathrm{vol}(\hat{E}') = \mathrm{vol}(B(0,1)) \cdot |\det(\hat{L}')| \ ,$$

  where $\hat{Q}' = \hat{L}'\hat{L}'^t$.

- We have

  $$\hat{L}'^{-1} = \begin{pmatrix} \frac{1}{a} & 0 & \dots & 0 \\ 0 & \frac{1}{b} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \frac{1}{b} \end{pmatrix} \text{ and } \hat{L}' = \begin{pmatrix} a & 0 & \dots & 0 \\ 0 & b & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & b \end{pmatrix}$$

- Note that a and b in the above equations depend on t, by the previous equations.

The Easy Case

- We want to choose t such that the volume of $\hat{E}'$ is minimal.

$$\mathrm{vol}(\hat{E}') = \mathrm{vol}(B(0,1)) \cdot |\det(\hat{L}')| \enspace ,$$

  where $\hat{Q}' = \hat{L}'\hat{L}'^t$.

- We have

$$\hat{L}'^{-1} = \begin{pmatrix} \frac{1}{a} & 0 & \dots & 0 \\ 0 & \frac{1}{b} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \frac{1}{b} \end{pmatrix} \text{ and } \hat{L}' = \begin{pmatrix} a & 0 & \dots & 0 \\ 0 & b & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & b \end{pmatrix}$$

- Note that a and b in the above equations depend on t, by the previous equations.

The Easy Case

- We want to choose t such that the volume of $\hat{E}'$ is minimal.

$$\mathrm{vol}(\hat{E}') = \mathrm{vol}(B(0,1)) \cdot |\det(\hat{L}')| \ ,$$

  where $\hat{Q}' = \hat{L}'\hat{L}'^t$.

- We have

$$\hat{L}'^{-1} = \begin{pmatrix} \frac{1}{a} & 0 & \dots & 0 \\ 0 & \frac{1}{b} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \frac{1}{b} \end{pmatrix} \text{ and } \hat{L}' = \begin{pmatrix} a & 0 & \dots & 0 \\ 0 & b & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & b \end{pmatrix}$$

- Note that a and b in the above equations depend on t, by the previous equations.

The Easy Case

$$\mathrm{vol}(\hat{E}')$$

The Easy Case

$$\mathrm{vol}(\hat{E}') = \mathrm{vol}(B(0,1)) \cdot |\det(\hat{L}')|$$

The Easy Case

$$\begin{aligned}\mathrm{vol}(\hat{E}') &= \mathrm{vol}(B(0,1)) \cdot |\det(\hat{L}')| \\ &= \mathrm{vol}(B(0,1)) \cdot ab^{n-1}\end{aligned}$$

The Easy Case

$$\begin{aligned}\mathrm{vol}(\hat{E}') &= \mathrm{vol}(B(0,1)) \cdot |\det(\hat{L}')| \\ &= \mathrm{vol}(B(0,1)) \cdot ab^{n-1} \\ &= \mathrm{vol}(B(0,1)) \cdot (1-t) \cdot \left(\frac{1-t}{\sqrt{1-2t}}\right)^{n-1}\end{aligned}$$

The Easy Case

$$\begin{aligned}\mathrm{vol}(\hat{E}') &= \mathrm{vol}(B(0,1)) \cdot |\det(\hat{L}')| \\ &= \mathrm{vol}(B(0,1)) \cdot ab^{n-1} \\ &= \mathrm{vol}(B(0,1)) \cdot (1-t) \cdot \left(\frac{1-t}{\sqrt{1-2t}}\right)^{n-1} \\ &= \mathrm{vol}(B(0,1)) \cdot \frac{(1-t)^n}{(\sqrt{1-2t})^{n-1}}\end{aligned}$$

The Easy Case

$$\frac{\mathrm{d}\,\mathrm{vol}(\hat{E}')}{\mathrm{d}\,t}$$

The Easy Case

$$\frac{\mathrm{d}\,\mathrm{vol}(\hat{E}')}{\mathrm{d}\,t} = \frac{\mathrm{d}}{\mathrm{d}\,t}\left(\frac{(1-t)^n}{(\sqrt{1-2t})^{n-1}}\right)$$

The Easy Case

$$\frac{\mathrm{d}\,\mathrm{vol}(\hat{E}')}{\mathrm{d}\,t} = \frac{\mathrm{d}}{\mathrm{d}\,t}\left(\frac{(1-t)^n}{(\sqrt{1-2t})^{n-1}}\right)$$

$$= \frac{1}{N^2}$$

N = denominator

The Easy Case

$$
\begin{aligned}
\frac{\mathrm{d}\,\mathrm{vol}(\hat{E}')}{\mathrm{d}\,t} &= \frac{\mathrm{d}}{\mathrm{d}\,t}\left(\frac{(1-t)^n}{(\sqrt{1-2t})^{n-1}}\right)\\
&= \frac{1}{N^2}\cdot\Bigg((-1)\cdot n(1-t)^{n-1}
\end{aligned}
$$

derivative of numerator

The Easy Case

$$
\begin{aligned}
\frac{\mathrm{d}\,\mathrm{vol}(\hat{E}')}{\mathrm{d}\,t} &= \frac{\mathrm{d}}{\mathrm{d}\,t}\left(\frac{(1-t)^n}{(\sqrt{1-2t})^{n-1}}\right)\\
&= \frac{1}{N^2}\cdot\Bigg((-1)\cdot n(1-t)^{n-1}\cdot \underset{\boxed{\text{denominator}}}{(\sqrt{1-2t})^{n-1}}
\end{aligned}
$$

The Easy Case

$$\begin{aligned}\frac{\mathrm{d}\,\mathrm{vol}(\hat{E}')}{\mathrm{d}\,t} &= \frac{\mathrm{d}}{\mathrm{d}\,t}\left(\frac{(1-t)^n}{(\sqrt{1-2t})^{n-1}}\right)\\ &= \frac{1}{N^2}\cdot\Bigg((-1)\cdot n(1-t)^{n-1}\cdot(\sqrt{1-2t})^{n-1}\\ &\quad -(n-1)(\sqrt{1-2t})^{n-2}\end{aligned}$$

outer derivative

The Easy Case

$$
\begin{aligned}
\frac{\mathrm{d}\,\mathrm{vol}(\hat{E}')}{\mathrm{d}\,t} &= \frac{\mathrm{d}}{\mathrm{d}\,t}\left(\frac{(1-t)^n}{(\sqrt{1-2t})^{n-1}}\right) \\
&= \frac{1}{N^2}\cdot\Bigg((-1)\cdot n(1-t)^{n-1}\cdot(\sqrt{1-2t})^{n-1} \\
&\quad -(n-1)(\sqrt{1-2t})^{n-2}\cdot \underbrace{\frac{1}{2\sqrt{1-2t}}\cdot(-2)}_{\text{inner derivative}}
\end{aligned}
$$

The Easy Case

$$
\begin{aligned}
\frac{\mathrm{d}\,\mathrm{vol}(\hat{E}')}{\mathrm{d}\,t} &= \frac{\mathrm{d}}{\mathrm{d}\,t}\left(\frac{(1-t)^n}{(\sqrt{1-2t})^{n-1}}\right)\\
&= \frac{1}{N^2}\cdot\Bigg((-1)\cdot n(1-t)^{n-1}\cdot(\sqrt{1-2t})^{n-1}\\
&\qquad -(n-1)(\sqrt{1-2t})^{n-2}\cdot\frac{1}{2\sqrt{1-2t}}\cdot(-2)\cdot\underset{\boxed{\text{numerator}}}{(1-t)^n}\Bigg)
\end{aligned}
$$

The Easy Case

$$\begin{aligned}
\frac{\mathrm{d}\,\mathrm{vol}(\hat{E}')}{\mathrm{d}\,t} &= \frac{\mathrm{d}}{\mathrm{d}\,t}\left(\frac{(1-t)^n}{(\sqrt{1-2t})^{n-1}}\right) \\
&= \frac{1}{N^2}\cdot\Bigg((-1)\cdot n(1-t)^{n-1}\cdot(\sqrt{1-2t})^{n-1} \\
&\quad -(n-1)(\sqrt{1-2t})^{n-2}\cdot\frac{1}{2\sqrt{1-2t}}\cdot(-2)\cdot(1-t)^n\Bigg) \\
&= \frac{1}{N^2}\cdot(\sqrt{1-2t})^{n-3}\cdot(1-t)^{n-1}
\end{aligned}$$

The Easy Case

$$
\begin{aligned}
\frac{\mathrm{d}\,\mathrm{vol}(\hat{E}')}{\mathrm{d}\,t} &= \frac{\mathrm{d}}{\mathrm{d}\,t}\left(\frac{(1-t)^n}{(\sqrt{1-2t})^{n-1}}\right) \\
&= \frac{1}{N^2}\cdot\Bigg((-1)\cdot n(1-t)^{n-1}\cdot \overset{1-2t}{\cancel{(\sqrt{1-2t})^{n-1}}} \\
&\quad -(n-1)(\sqrt{1-2t})^{n-2}\cdot\frac{1}{2\sqrt{1-2t}}\cdot(-2)\cdot(1-t)^n\Bigg) \\
&= \frac{1}{N^2}\cdot(\sqrt{1-2t})^{n-3}\cdot(1-t)^{n-1}
\end{aligned}
$$

The Easy Case

$$
\begin{aligned}
\frac{\mathrm{d}\,\mathrm{vol}(\hat{E}')}{\mathrm{d}\,t} &= \frac{\mathrm{d}}{\mathrm{d}\,t}\left(\frac{(1-t)^n}{(\sqrt{1-2t})^{n-1}}\right) \\
&= \frac{1}{N^2}\cdot\Bigg((-1)\cdot n(1-t)^{n-1}\cdot \overset{1-2t}{\cancel{(\sqrt{1-2t})^{n-1}}} \\
&\qquad -(n-1)\cancel{(\sqrt{1-2t})^{n-2}}\cdot\frac{1}{2\cancel{\sqrt{1-2t}}}\cdot(-2)\cdot(1-t)^n\Bigg) \\
&= \frac{1}{N^2}\cdot(\sqrt{1-2t})^{n-3}\cdot(1-t)^{n-1}
\end{aligned}
$$

The Easy Case

$$
\begin{aligned}
\frac{\mathrm{d}\,\mathrm{vol}(\hat{E}')}{\mathrm{d}\,t} &= \frac{\mathrm{d}}{\mathrm{d}\,t}\left(\frac{(1-t)^n}{(\sqrt{1-2t})^{n-1}}\right)\\
&= \frac{1}{N^2}\cdot\Bigg((-1)\cdot n\cancel{(1-t)^{n-1}}\cdot\overset{1-2t}{\cancel{(\sqrt{1-2t})^{n-1}}}\\
&\qquad -(n-1)\cancel{(\sqrt{1-2t})^{n-2}}\cdot\frac{1}{2\cancel{\sqrt{1-2t}}}\cdot(-2)\cdot(1-t)^n\Bigg)\\
&= \frac{1}{N^2}\cdot(\sqrt{1-2t})^{n-3}\cdot(1-t)^{n-1}
\end{aligned}
$$

The Easy Case

$$
\begin{aligned}
\frac{\mathrm{d}\,\mathrm{vol}(\hat{E}')}{\mathrm{d}\,t} &= \frac{\mathrm{d}}{\mathrm{d}\,t}\left(\frac{(1-t)^n}{(\sqrt{1-2t})^{n-1}}\right)\\
&= \frac{1}{N^2}\cdot\Bigg((-1)\cdot n\cancel{(1-t)^{n-1}}\cdot\overset{1-2t}{\cancel{(\sqrt{1-2t})^{n-1}}}\\
&\qquad -(n-1)\cancel{(\sqrt{1-2t})^{n-2}}\cdot\frac{1}{2\cancel{\sqrt{1-2t}}}\cdot(-2)\cdot\overset{1-t}{\cancel{(1-t)^n}}\Bigg)\\
&= \frac{1}{N^2}\cdot(\sqrt{1-2t})^{n-3}\cdot(1-t)^{n-1}
\end{aligned}
$$

The Easy Case

$$
\begin{aligned}
\frac{\mathrm{d}\,\mathrm{vol}(\hat{E}')}{\mathrm{d}\,t} &= \frac{\mathrm{d}}{\mathrm{d}\,t}\left(\frac{(1-t)^n}{(\sqrt{1-2t})^{n-1}}\right) \\
&= \frac{1}{N^2}\cdot\Bigg((-1)\cdot n\cancel{(1-t)^{n-1}}\cdot\overset{1-2t}{\cancel{(\sqrt{1-2t})^{n-1}}} \\
&\qquad \cancel{-}(n-1)\cancel{(\sqrt{1-2t})^{n-2}}\cdot\frac{1}{\cancel{2}\cancel{\sqrt{1-2t}}}\cdot\cancel{(-2)}\cdot\overset{1-t}{\cancel{(1-t)^n}}\Bigg) \\
&= \frac{1}{N^2}\cdot(\sqrt{1-2t})^{n-3}\cdot(1-t)^{n-1}
\end{aligned}
$$

The Easy Case

$$
\begin{aligned}
\frac{\mathrm{d}\,\mathrm{vol}(\hat{E}')}{\mathrm{d}\,t} &= \frac{\mathrm{d}}{\mathrm{d}\,t}\left(\frac{(1-t)^n}{(\sqrt{1-2t})^{n-1}}\right) \\
&= \frac{1}{N^2}\cdot\Bigg((-1)\cdot n\cancel{(1-t)^{n-1}}\cdot\overset{1-2t}{\cancel{(\sqrt{1-2t})^{n-1}}} \\
&\quad \cancel{-}(n-1)\cancel{(\sqrt{1-2t})^{n-2}}\cdot\frac{1}{\cancel{2}\cancel{\sqrt{1-2t}}}\cdot\cancel{(-2)}\cdot\overset{1-t}{\cancel{(1-t)^n}}\Bigg) \\
&= \frac{1}{N^2}\cdot(\sqrt{1-2t})^{n-3}\cdot(1-t)^{n-1} \\
&\qquad\cdot\Big((n-1)(1-t)-n(1-2t)\Big)
\end{aligned}
$$

The Easy Case

$$
\begin{aligned}
\frac{\mathrm{d}\,\mathrm{vol}(\hat{E}')}{\mathrm{d}\,t} &= \frac{\mathrm{d}}{\mathrm{d}\,t}\left(\frac{(1-t)^n}{(\sqrt{1-2t})^{n-1}}\right)\\
&= \frac{1}{N^2}\cdot\Bigg((-1)\cdot n\cancel{(1-t)^{n-1}}\cdot\overset{1-2t}{\cancel{(\sqrt{1-2t})^{n-1}}}\\
&\quad \cancel{-}(n-1)\cancel{(\sqrt{1-2t})^{n-2}}\cdot\frac{1}{\cancel{2}\cancel{\sqrt{1-2t}}}\cdot(\cancel{-2})\cdot\overset{1-t}{\cancel{(1-t)^n}}\Bigg)\\
&= \frac{1}{N^2}\cdot(\sqrt{1-2t})^{n-3}\cdot(1-t)^{n-1}\\
&\qquad\cdot\Big((n-1)(1-t)-n(1-2t)\Big)\\
&= \frac{1}{N^2}\cdot(\sqrt{1-2t})^{n-3}\cdot(1-t)^{n-1}\cdot\Big((n+1)t-1\Big)
\end{aligned}
$$

The Easy Case

- We obtain the minimum for $t = \frac{1}{n+1}$.
- For this value we obtain

$$a$$

The Easy Case

- We obtain the minimum for $t = \frac{1}{n+1}$.
- For this value we obtain

$$a = 1 - t$$

The Easy Case

- We obtain the minimum for $t = \frac{1}{n+1}$.
- For this value we obtain

$$a = 1 - t = \frac{n}{n+1}$$

The Easy Case

- We obtain the minimum for $t = \frac{1}{n+1}$.
- For this value we obtain

$$a = 1 - t = \frac{n}{n+1} \text{ and } b =$$

The Easy Case

- We obtain the minimum for $t = \frac{1}{n+1}$.
- For this value we obtain

$$a = 1 - t = \frac{n}{n+1} \text{ and } b = \frac{1-t}{\sqrt{1-2t}}$$

The Easy Case

- We obtain the minimum for $t = \frac{1}{n+1}$.
- For this value we obtain

$$a = 1 - t = \frac{n}{n+1} \text{ and } b = \frac{1-t}{\sqrt{1-2t}} = \frac{n}{\sqrt{n^2-1}}$$

The Easy Case

- We obtain the minimum for $t = \frac{1}{n+1}$.
- For this value we obtain

$$a = 1 - t = \frac{n}{n+1} \text{ and } b = \frac{1-t}{\sqrt{1-2t}} = \frac{n}{\sqrt{n^2-1}}$$

To see the equation for b, observe that

$$b^2$$

The Easy Case

- We obtain the minimum for $t = \frac{1}{n+1}$.
- For this value we obtain

$$a = 1 - t = \frac{n}{n+1} \text{ and } b = \frac{1-t}{\sqrt{1-2t}} = \frac{n}{\sqrt{n^2-1}}$$

To see the equation for b, observe that

$$b^2 = \frac{(1-t)^2}{1-2t}$$

The Easy Case

- We obtain the minimum for $t = \frac{1}{n+1}$.
- For this value we obtain

$$a = 1 - t = \frac{n}{n+1} \text{ and } b = \frac{1-t}{\sqrt{1-2t}} = \frac{n}{\sqrt{n^2-1}}$$

To see the equation for b, observe that

$$b^2 = \frac{(1-t)^2}{1-2t} = \frac{(1-\frac{1}{n+1})^2}{1-\frac{2}{n+1}}$$

The Easy Case

- We obtain the minimum for $t = \frac{1}{n+1}$.
- For this value we obtain

$$a = 1 - t = \frac{n}{n+1} \text{ and } b = \frac{1-t}{\sqrt{1-2t}} = \frac{n}{\sqrt{n^2-1}}$$

To see the equation for b, observe that

$$b^2 = \frac{(1-t)^2}{1-2t} = \frac{(1-\frac{1}{n+1})^2}{1-\frac{2}{n+1}} = \frac{(\frac{n}{n+1})^2}{\frac{n-1}{n+1}}$$

The Easy Case

- We obtain the minimum for $t = \frac{1}{n+1}$.
- For this value we obtain

$$a = 1 - t = \frac{n}{n+1} \text{ and } b = \frac{1-t}{\sqrt{1-2t}} = \frac{n}{\sqrt{n^2-1}}$$

To see the equation for b, observe that

$$b^2 = \frac{(1-t)^2}{1-2t} = \frac{(1-\frac{1}{n+1})^2}{1-\frac{2}{n+1}} = \frac{(\frac{n}{n+1})^2}{\frac{n-1}{n+1}} = \frac{n^2}{n^2-1}$$

The Easy Case

Let $\gamma_n = \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(B(0,1))} = ab^{n-1}$ be the ratio by which the volume changes:

$$\gamma_n^2$$

The Easy Case

Let $\gamma_n = \frac{\text{vol}(\hat{E}')}{\text{vol}(B(0,1))} = ab^{n-1}$ be the ratio by which the volume changes:

$$\gamma_n^2 = \Big(\frac{n}{n+1}\Big)^2\Big(\frac{n^2}{n^2-1}\Big)^{n-1}$$

The Easy Case

Let $\gamma_n = \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(B(0,1))} = ab^{n-1}$ be the ratio by which the volume changes:

$$\begin{aligned}\gamma_n^2 &= \Big(\frac{n}{n+1}\Big)^2\Big(\frac{n^2}{n^2-1}\Big)^{n-1}\\ &= \Big(1-\frac{1}{n+1}\Big)^2\Big(1+\frac{1}{(n-1)(n+1)}\Big)^{n-1}\end{aligned}$$

The Easy Case

Let $\gamma_n = \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(B(0,1))} = ab^{n-1}$ be the ratio by which the volume changes:

$$\begin{aligned}
\gamma_n^2 &= \Big(\frac{n}{n+1}\Big)^2\Big(\frac{n^2}{n^2-1}\Big)^{n-1} \\
&= \Big(1 - \frac{1}{n+1}\Big)^2\Big(1 + \frac{1}{(n-1)(n+1)}\Big)^{n-1} \\
&\le e^{-2\frac{1}{n+1}} \cdot e^{\frac{1}{n+1}}
\end{aligned}$$

The Easy Case

Let $\gamma_n = \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(B(0,1))} = ab^{n-1}$ be the ratio by which the volume changes:

$$\begin{aligned}
\gamma_n^2 &= \Big(\frac{n}{n+1}\Big)^2\Big(\frac{n^2}{n^2-1}\Big)^{n-1} \\
&= \Big(1-\frac{1}{n+1}\Big)^2\Big(1+\frac{1}{(n-1)(n+1)}\Big)^{n-1} \\
&\le e^{-2\frac{1}{n+1}}\cdot e^{\frac{1}{n+1}} \\
&= e^{-\frac{1}{n+1}}
\end{aligned}$$

The Easy Case

Let $\gamma_n = \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(B(0,1))} = ab^{n-1}$ be the ratio by which the volume changes:

$$\begin{aligned}\gamma_n^2 &= \Big(\frac{n}{n+1}\Big)^2\Big(\frac{n^2}{n^2-1}\Big)^{n-1}\\ &= \Big(1-\frac{1}{n+1}\Big)^2\Big(1+\frac{1}{(n-1)(n+1)}\Big)^{n-1}\\ &\le e^{-2\frac{1}{n+1}}\cdot e^{\frac{1}{n+1}}\\ &= e^{-\frac{1}{n+1}}\end{aligned}$$

where we used $(1+x)^a \le e^{ax}$ for $x \in \mathbb{R}$ and $a > 0$.

The Easy Case

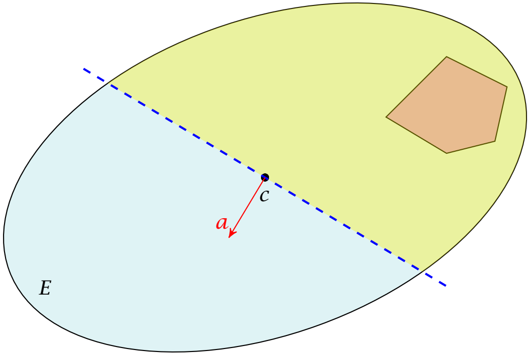
Let $\gamma_n = \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(B(0,1))} = ab^{n-1}$ be the ratio by which the volume changes:

$$\begin{aligned}\gamma_n^2 &= \Big(\frac{n}{n+1}\Big)^2\Big(\frac{n^2}{n^2-1}\Big)^{n-1}\\ &= \Big(1-\frac{1}{n+1}\Big)^2\Big(1+\frac{1}{(n-1)(n+1)}\Big)^{n-1}\\ &\le e^{-2\frac{1}{n+1}}\cdot e^{\frac{1}{n+1}}\\ &= e^{-\frac{1}{n+1}}\end{aligned}$$

where we used $(1+x)^a \le e^{ax}$ for $x\in\mathbb{R}$ and $a>0$.

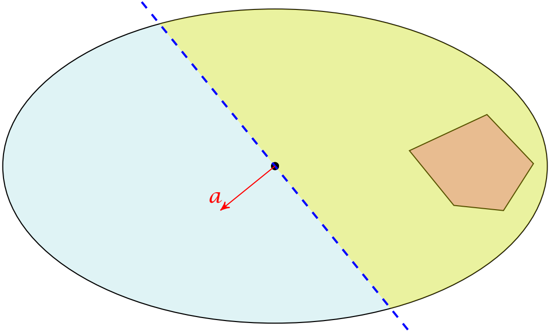
This gives $\gamma_n \le e^{-\frac{1}{2(n+1)}}$.

How to Compute the New Ellipsoid



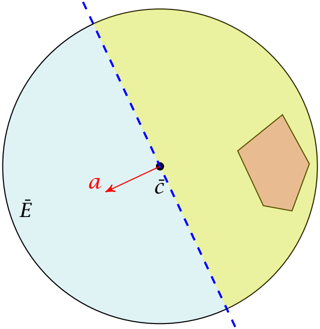
How to Compute the New Ellipsoid

- Use f^{-1} (recall that $f = Lx + t$ is the affine transformation of the unit ball) to rotate/distort the ellipsoid (back) into the unit ball.



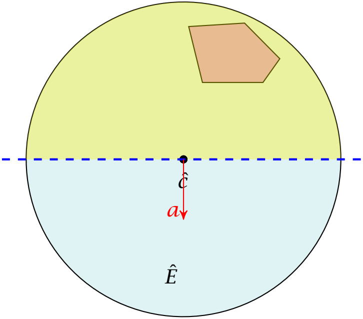
How to Compute the New Ellipsoid

- Use f^{-1} (recall that $f = Lx + t$ is the affine transformation of the unit ball) to rotate/distort the ellipsoid (back) into the unit ball.



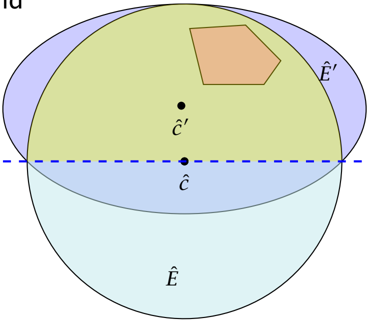
How to Compute the New Ellipsoid

- Use f^{-1} (recall that $f = Lx + t$ is the affine transformation of the unit ball) to rotate/distort the ellipsoid (back) into the unit ball.
- Use a rotation R^{-1} to rotate the unit ball such that the normal vector of the halfspace is parallel to e_1.



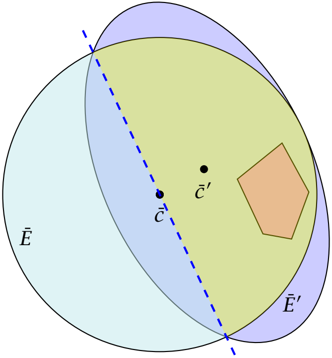
How to Compute the New Ellipsoid

- Use f^{-1} (recall that $f = Lx + t$ is the affine transformation of the unit ball) to rotate/distort the ellipsoid (back) into the unit ball.
- Use a rotation R^{-1} to rotate the unit ball such that the normal vector of the halfspace is parallel to e_1.
- Compute the new center $\hat{c}'$ and the new matrix $\hat{Q}'$ for this simplified setting.



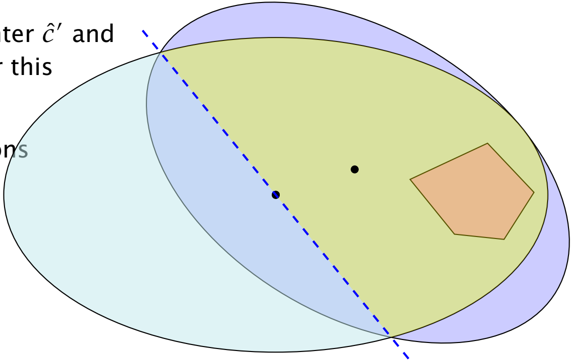
How to Compute the New Ellipsoid

- Use f^{-1} (recall that $f = Lx + t$ is the affine transformation of the unit ball) to rotate/distort the ellipsoid (back) into the unit ball.
- Use a rotation R^{-1} to rotate the unit ball such that the normal vector of the halfspace is parallel to e_1.
- Compute the new center $\hat{c}'$ and the new matrix $\hat{Q}'$ for this simplified setting.
- Use the transformations R and f to get the new center c' and the new matrix Q' for the original ellipsoid E.



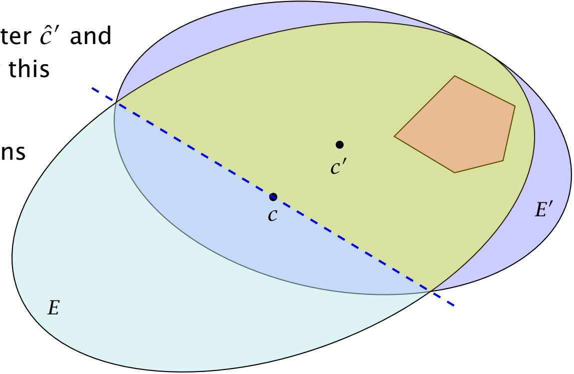
How to Compute the New Ellipsoid

- Use f^{-1} (recall that $f = Lx + t$ is the affine transformation of the unit ball) to rotate/distort the ellipsoid (back) into the unit ball.
- Use a rotation R^{-1} to rotate the unit ball such that the normal vector of the halfspace is parallel to e_1.
- Compute the new center $\hat{c}'$ and the new matrix $\hat{Q}'$ for this simplified setting.
- Use the transformations R and f to get the new center c' and the new matrix Q' for the original ellipsoid E.



How to Compute the New Ellipsoid

- Use f^{-1} (recall that $f = Lx + t$ is the affine transformation of the unit ball) to rotate/distort the ellipsoid (back) into the unit ball.
- Use a rotation R^{-1} to rotate the unit ball such that the normal vector of the halfspace is parallel to e_1.
- Compute the new center $\hat{c}'$ and the new matrix $\hat{Q}'$ for this simplified setting.
- Use the transformations R and f to get the new center c' and the new matrix Q' for the original ellipsoid E.



Our progress is the same:

$$e^{-\frac{1}{2(n+1)}}$$

Our progress is the same:

$$e^{-\frac{1}{2(n+1)}} \geq \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(B(0,1))}$$

Our progress is the same:

$$e^{-\frac{1}{2(n+1)}} \geq \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(B(0,1))} = \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(\hat{E})}$$

Our progress is the same:

$$e^{-\frac{1}{2(n+1)}} \geq \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(B(0,1))} = \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(\hat{E})} = \frac{\mathrm{vol}(R(\hat{E}'))}{\mathrm{vol}(R(\hat{E}))}$$

Our progress is the same:

$$\begin{aligned} e^{-\frac{1}{2(n+1)}} &\geq \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(B(0,1))} = \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(\hat{E})} = \frac{\mathrm{vol}(R(\hat{E}'))}{\mathrm{vol}(R(\hat{E}))} \\ &= \frac{\mathrm{vol}(\bar{E}')}{\mathrm{vol}(\bar{E})} \end{aligned}$$

Our progress is the same:

$$
\begin{aligned}
e^{-\frac{1}{2(n+1)}} \geq \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(B(0,1))} = \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(\hat{E})} = \frac{\mathrm{vol}(R(\hat{E}'))}{\mathrm{vol}(R(\hat{E}))} \\
= \frac{\mathrm{vol}(\bar{E}')}{\mathrm{vol}(\bar{E})} = \frac{\mathrm{vol}(f(\bar{E}'))}{\mathrm{vol}(f(\bar{E}))}
\end{aligned}
$$

Our progress is the same:

$$e^{-\frac{1}{2(n+1)}} \geq \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(B(0,1))} = \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(\hat{E})} = \frac{\mathrm{vol}(R(\hat{E}'))}{\mathrm{vol}(R(\hat{E}))}$$

$$= \frac{\mathrm{vol}(\bar{E}')}{\mathrm{vol}(\bar{E})} = \frac{\mathrm{vol}(f(\bar{E}'))}{\mathrm{vol}(f(\bar{E}))} = \frac{\mathrm{vol}(E')}{\mathrm{vol}(E)}$$

Our progress is the same:

$$
\begin{aligned}
e^{-\frac{1}{2(n+1)}} \geq \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(B(0,1))} &= \frac{\mathrm{vol}(\hat{E}')}{\mathrm{vol}(\hat{E})} = \frac{\mathrm{vol}(R(\hat{E}'))}{\mathrm{vol}(R(\hat{E}))} \\
&= \frac{\mathrm{vol}(\bar{E}')}{\mathrm{vol}(\bar{E})} = \frac{\mathrm{vol}(f(\bar{E}'))}{\mathrm{vol}(f(\bar{E}))} = \frac{\mathrm{vol}(E')}{\mathrm{vol}(E)}
\end{aligned}
$$

Here it is important that mapping a set with affine function $f(x) = Lx + t$ changes the volume by factor $\det(L)$.

The Ellipsoid Algorithm

How to Compute The New Parameters?

The Ellipsoid Algorithm

How to Compute The New Parameters?

The transformation function of the (old) ellipsoid: $f(x) = Lx + c$;

The Ellipsoid Algorithm

How to Compute The New Parameters?

The transformation function of the (old) ellipsoid: $f(x) = Lx + c$;

The halfspace to be intersected: $H = \{x \mid a^t(x - c) \leq 0\}$;

The Ellipsoid Algorithm

How to Compute The New Parameters?

The transformation function of the (old) ellipsoid: $f(x) = Lx + c$;

The halfspace to be intersected: $H = \{x \mid a^t(x - c) \leq 0\}$;

$$f^{-1}(H) = \{f^{-1}(x) \mid a^t(x - c) \leq 0\}$$

The Ellipsoid Algorithm

How to Compute The New Parameters?

The transformation function of the (old) ellipsoid: $f(x) = Lx + c$;

The halfspace to be intersected: $H = \{x \mid a^t(x - c) \le 0\}$;

$$
\begin{aligned}
f^{-1}(H) &= \{f^{-1}(x) \mid a^t(x - c) \le 0\} \\
&= \{f^{-1}(f(y)) \mid a^t(f(y) - c) \le 0\}
\end{aligned}
$$

The Ellipsoid Algorithm

How to Compute The New Parameters?

The transformation function of the (old) ellipsoid: $f(x) = Lx + c$;

The halfspace to be intersected: $H = \{x \mid a^t(x - c) \leq 0\}$;

$$\begin{aligned} f^{-1}(H) &= \{f^{-1}(x) \mid a^t(x - c) \leq 0\} \\ &= \{f^{-1}(f(y)) \mid a^t(f(y) - c) \leq 0\} \\ &= \{y \mid a^t(f(y) - c) \leq 0\} \end{aligned}$$

The Ellipsoid Algorithm

How to Compute The New Parameters?

The transformation function of the (old) ellipsoid: $f(x) = Lx + c$;

The halfspace to be intersected: $H = \{x \mid a^t(x - c) \leq 0\}$;

$$\begin{aligned} f^{-1}(H) &= \{f^{-1}(x) \mid a^t(x - c) \leq 0\} \\ &= \{f^{-1}(f(y)) \mid a^t(f(y) - c) \leq 0\} \\ &= \{y \mid a^t(f(y) - c) \leq 0\} \\ &= \{y \mid a^t(Ly + c - c) \leq 0\} \end{aligned}$$

The Ellipsoid Algorithm

How to Compute The New Parameters?

The transformation function of the (old) ellipsoid: $f(x) = Lx + c$;

The halfspace to be intersected: $H = \{x \mid a^t(x - c) \le 0\}$;

$$\begin{aligned}
f^{-1}(H) &= \{f^{-1}(x) \mid a^t(x - c) \le 0\} \\
&= \{f^{-1}(f(y)) \mid a^t(f(y) - c) \le 0\} \\
&= \{y \mid a^t(f(y) - c) \le 0\} \\
&= \{y \mid a^t(Ly + c - c) \le 0\} \\
&= \{y \mid (a^tL)y \le 0\}
\end{aligned}$$

The Ellipsoid Algorithm

How to Compute The New Parameters?

The transformation function of the (old) ellipsoid: $f(x) = Lx + c$;

The halfspace to be intersected: $H = \{x \mid a^t(x - c) \le 0\}$;

$$\begin{aligned} f^{-1}(H) &= \{f^{-1}(x) \mid a^t(x - c) \le 0\} \\ &= \{f^{-1}(f(y)) \mid a^t(f(y) - c) \le 0\} \\ &= \{y \mid a^t(f(y) - c) \le 0\} \\ &= \{y \mid a^t(Ly + c - c) \le 0\} \\ &= \{y \mid (a^tL)y \le 0\} \end{aligned}$$

This means $\bar{a} = L^t a$.

The Ellipsoid Algorithm

After rotating back (applying R^{-1}) the normal vector of the halfspace points in negative x_1-direction. Hence,

$$R^{-1}\Big(\frac{L^t a}{\|L^t a\|}\Big) = -e_1 \quad \Rightarrow \quad -\frac{L^t a}{\|L^t a\|} = R \cdot e_1$$

The Ellipsoid Algorithm

After rotating back (applying R^{-1}) the normal vector of the halfspace points in negative x_1-direction. Hence,

$$R^{-1}\Big(\frac{L^t a}{\|L^t a\|}\Big) = -e_1 \quad \Rightarrow \quad -\frac{L^t a}{\|L^t a\|} = R \cdot e_1$$

Hence,

$$\bar{c}'$$

The Ellipsoid Algorithm

After rotating back (applying R^{-1}) the normal vector of the halfspace points in negative x_1-direction. Hence,

$$R^{-1}\Big(\frac{L^t a}{\|L^t a\|}\Big) = -e_1 \quad \Rightarrow \quad -\frac{L^t a}{\|L^t a\|} = R \cdot e_1$$

Hence,

$$\bar{c}' = R \cdot \hat{c}'$$

The Ellipsoid Algorithm

After rotating back (applying R^{-1}) the normal vector of the halfspace points in negative x_1-direction. Hence,

$$R^{-1}\Big(\frac{L^t a}{\|L^t a\|}\Big) = -e_1 \quad \Rightarrow \quad -\frac{L^t a}{\|L^t a\|} = R \cdot e_1$$

Hence,

$$\bar{c}' = R \cdot \hat{c}' = R \cdot \frac{1}{n+1} e_1$$

The Ellipsoid Algorithm

After rotating back (applying R^{-1}) the normal vector of the halfspace points in negative x_1-direction. Hence,

$$R^{-1}\Big(\frac{L^t a}{\|L^t a\|}\Big) = -e_1 \quad \Rightarrow \quad -\frac{L^t a}{\|L^t a\|} = R \cdot e_1$$

Hence,

$$\bar{c}' = R \cdot \hat{c}' = R \cdot \frac{1}{n+1} e_1 = -\frac{1}{n+1} \frac{L^t a}{\|L^t a\|}$$

The Ellipsoid Algorithm

After rotating back (applying R^{-1}) the normal vector of the halfspace points in negative x_1-direction. Hence,

$$R^{-1}\Big(\frac{L^t a}{\|L^t a\|}\Big) = -e_1 \quad\Rightarrow\quad -\frac{L^t a}{\|L^t a\|} = R\cdot e_1$$

Hence,

$$\bar{c}' = R\cdot\hat{c}' = R\cdot\frac{1}{n+1}e_1 = -\frac{1}{n+1}\frac{L^t a}{\|L^t a\|}$$

c'

The Ellipsoid Algorithm

After rotating back (applying R^{-1}) the normal vector of the halfspace points in negative x_1-direction. Hence,

$$R^{-1}\Big(\frac{L^t a}{\|L^t a\|}\Big) = -e_1 \quad \Rightarrow \quad -\frac{L^t a}{\|L^t a\|} = R \cdot e_1$$

Hence,

$$\bar{c}' = R \cdot \hat{c}' = R \cdot \frac{1}{n+1} e_1 = -\frac{1}{n+1}\frac{L^t a}{\|L^t a\|}$$

$$c' = f(\bar{c}')$$

The Ellipsoid Algorithm

After rotating back (applying R^{-1}) the normal vector of the halfspace points in negative x_1-direction. Hence,

$$R^{-1}\Big(\frac{L^t a}{\|L^t a\|}\Big) = -e_1 \quad \Rightarrow \quad -\frac{L^t a}{\|L^t a\|} = R \cdot e_1$$

Hence,

$$\bar{c}' = R \cdot \hat{c}' = R \cdot \frac{1}{n+1} e_1 = -\frac{1}{n+1} \frac{L^t a}{\|L^t a\|}$$

$$c' = f(\bar{c}') = L \cdot \bar{c}' + c$$

The Ellipsoid Algorithm

After rotating back (applying R^{-1}) the normal vector of the halfspace points in negative x_1-direction. Hence,

$$R^{-1}\Big(\frac{L^t a}{\|L^t a\|}\Big) = -e_1 \quad \Rightarrow \quad -\frac{L^t a}{\|L^t a\|} = R \cdot e_1$$

Hence,

$$\bar{c}' = R \cdot \hat{c}' = R \cdot \frac{1}{n+1} e_1 = -\frac{1}{n+1} \frac{L^t a}{\|L^t a\|}$$

$$\begin{aligned} c' &= f(\bar{c}') = L \cdot \bar{c}' + c \\ &= -\frac{1}{n+1} L \frac{L^t a}{\|L^t a\|} + c \end{aligned}$$

The Ellipsoid Algorithm

After rotating back (applying R^{-1}) the normal vector of the halfspace points in negative x_1-direction. Hence,

$$R^{-1}\Big(\frac{L^t a}{\|L^t a\|}\Big) = -e_1 \quad \Rightarrow \quad -\frac{L^t a}{\|L^t a\|} = R \cdot e_1$$

Hence,

$$\bar{c}' = R \cdot \hat{c}' = R \cdot \frac{1}{n+1} e_1 = -\frac{1}{n+1} \frac{L^t a}{\|L^t a\|}$$

$$\begin{aligned} c' &= f(\bar{c}') = L \cdot \bar{c}' + c \\ &= -\frac{1}{n+1} L \frac{L^t a}{\|L^t a\|} + c \\ &= c - \frac{1}{n+1} \frac{Qa}{\sqrt{a^t Q a}} \end{aligned}$$

For computing the matrix Q' of the new ellipsoid we assume in the following that $\hat{E}', \bar{E}'$ and E' refer to the ellispoids centered in the origin.

Recall that

$$\hat{Q}' = \begin{pmatrix} a^2 & 0 & \dots & 0 \\ 0 & b^2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & b^2 \end{pmatrix}$$

This gives

$$\hat{Q}' = \frac{n^2}{n^2-1}\Big(I - \frac{2}{n+1}e_1e_1^t\Big)$$

because for $a = {}^{n}/_{n+1}$ and $b = {}^{n}/_{\sqrt{n^2-1}}$

$$\begin{aligned} b^2 - b^2\frac{2}{n+1} &= \frac{n^2}{n^2-1} - \frac{2n^2}{(n-1)(n+1)^2} \\ &= \frac{n^2(n+1) - 2n^2}{(n-1)(n+1)^2} = \frac{n^2(n-1)}{(n-1)(n+1)^2} = a^2 \end{aligned}$$

Recall that

$$\hat{Q}' = \begin{pmatrix} a^2 & 0 & \dots & 0 \\ 0 & b^2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & b^2 \end{pmatrix}$$

This gives

$$\hat{Q}' = \frac{n^2}{n^2-1}\Big(I - \frac{2}{n+1}e_1 e_1^t\Big)$$

because for $a = {}^{n}/_{n+1}$ and $b = {}^{n}/_{\sqrt{n^2-1}}$

$$b^2 - b^2\frac{2}{n+1} = \frac{n^2}{n^2-1} - \frac{2n^2}{(n-1)(n+1)^2}$$

$$= \frac{n^2(n+1) - 2n^2}{(n-1)(n+1)^2} = \frac{n^2(n-1)}{(n-1)(n+1)^2} = a^2$$

Recall that

$$\hat{Q}' = \begin{pmatrix} a^2 & 0 & \dots & 0 \\ 0 & b^2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & b^2 \end{pmatrix}$$

This gives

$$\hat{Q}' = \frac{n^2}{n^2-1}\Big(I - \frac{2}{n+1}e_1e_1^t\Big)$$

because for $a = n/_{n+1}$ and $b = n/_{\sqrt{n^2-1}}$

Recall that

$$\hat{Q}' = \begin{pmatrix} a^2 & 0 & \dots & 0 \\ 0 & b^2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & b^2 \end{pmatrix}$$

This gives

$$\hat{Q}' = \frac{n^2}{n^2-1}\Big(I - \frac{2}{n+1} e_1 e_1^t\Big)$$

because for $a = {}^{n}/_{n+1}$ and $b = {}^{n}/_{\sqrt{n^2-1}}$

$$\begin{aligned} b^2 - b^2\frac{2}{n+1} &= \frac{n^2}{n^2-1} - \frac{2n^2}{(n-1)(n+1)^2} \\ &= \frac{n^2(n+1)-2n^2}{(n-1)(n+1)^2} = \frac{n^2(n-1)}{(n-1)(n+1)^2} = a^2 \end{aligned}$$

Recall that

$$\hat{Q}' = \begin{pmatrix} a^2 & 0 & \dots & 0 \\ 0 & b^2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & b^2 \end{pmatrix}$$

This gives

$$\hat{Q}' = \frac{n^2}{n^2-1}\Big(I - \frac{2}{n+1}e_1e_1^t\Big)$$

because for $a = {}^{n}/_{n+1}$ and $b = {}^{n}/_{\sqrt{n^2-1}}$

$$b^2 - b^2\frac{2}{n+1} = \frac{n^2}{n^2-1} - \frac{2n^2}{(n-1)(n+1)^2}$$

$$= \frac{n^2(n+1) - 2n^2}{(n-1)(n+1)^2} = \frac{n^2(n-1)}{(n-1)(n+1)^2} = a^2$$

Recall that

$$\hat{Q}' = \begin{pmatrix} a^2 & 0 & \dots & 0 \\ 0 & b^2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & b^2 \end{pmatrix}$$

This gives

$$\hat{Q}' = \frac{n^2}{n^2-1}\Big(I - \frac{2}{n+1} e_1 e_1^t\Big)$$

because for $a = {}^{n}/_{n+1}$ and $b = {}^{n}/_{\sqrt{n^2-1}}$

$$\begin{aligned} b^2 - b^2\frac{2}{n+1} &= \frac{n^2}{n^2-1} - \frac{2n^2}{(n-1)(n+1)^2} \\ &= \frac{n^2(n+1) - 2n^2}{(n-1)(n+1)^2} = \frac{n^2(n-1)}{(n-1)(n+1)^2} = a^2 \end{aligned}$$

Recall that

$$\hat{Q}' = \begin{pmatrix} a^2 & 0 & \dots & 0 \\ 0 & b^2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & b^2 \end{pmatrix}$$

This gives

$$\hat{Q}' = \frac{n^2}{n^2-1}\Big(I - \frac{2}{n+1} e_1 e_1^t\Big)$$

because for $a = {}^{n}\!/_{n+1}$ and $b = {}^{n}\!/_{\sqrt{n^2-1}}$

$$\begin{aligned} b^2 - b^2 \frac{2}{n+1} &= \frac{n^2}{n^2-1} - \frac{2n^2}{(n-1)(n+1)^2} \\ &= \frac{n^2(n+1) - 2n^2}{(n-1)(n+1)^2} = \frac{n^2(n-1)}{(n-1)(n+1)^2} = a^2 \end{aligned}$$

Recall that

$$\hat{Q}' = \begin{pmatrix} a^2 & 0 & \dots & 0 \\ 0 & b^2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & b^2 \end{pmatrix}$$

This gives

$$\hat{Q}' = \frac{n^2}{n^2-1}\Big(I - \frac{2}{n+1} e_1 e_1^t\Big)$$

because for $a = {}^{n}/_{n+1}$ and $b = {}^{n}/_{\sqrt{n^2-1}}$

$$\begin{aligned} b^2 - b^2 \frac{2}{n+1} &= \frac{n^2}{n^2-1} - \frac{2n^2}{(n-1)(n+1)^2} \\ &= \frac{n^2(n+1) - 2n^2}{(n-1)(n+1)^2} = \frac{n^2(n-1)}{(n-1)(n+1)^2} = a^2 \end{aligned}$$

9 The Ellipsoid Algorithm

$\bar{E}'$

9 The Ellipsoid Algorithm

$$\bar{E}' = R(\hat{E}')$$

9 The Ellipsoid Algorithm

$$\begin{aligned}\bar{E}' &= R(\hat{E}') \\ &= \{R(x) \mid x^t {\hat{Q}'}^{-1} x \le 1\}\end{aligned}$$

9 The Ellipsoid Algorithm

$$
\begin{aligned}
\bar{E}' &= R(\hat{E}') \\
&= \{R(x) \mid x^t \hat{Q}'^{-1} x \le 1\} \\
&= \{y \mid (R^{-1}y)^t \hat{Q}'^{-1} R^{-1} y \le 1\}
\end{aligned}
$$

9 The Ellipsoid Algorithm

$$
\begin{aligned}
\bar{E}' &= R(\hat{E}') \\
&= \{R(x) \mid x^t \hat{Q}'^{-1} x \le 1\} \\
&= \{y \mid (R^{-1}y)^t \hat{Q}'^{-1} R^{-1} y \le 1\} \\
&= \{y \mid y^t (R^t)^{-1} \hat{Q}'^{-1} R^{-1} y \le 1\}
\end{aligned}
$$

9 The Ellipsoid Algorithm

$$
\begin{aligned}
\bar{E}' &= R(\hat{E}') \\
&= \{R(x) \mid x^t \hat{Q}'^{-1} x \le 1\} \\
&= \{y \mid (R^{-1}y)^t \hat{Q}'^{-1} R^{-1} y \le 1\} \\
&= \{y \mid y^t (R^t)^{-1} \hat{Q}'^{-1} R^{-1} y \le 1\} \\
&= \{y \mid y^t (\underbrace{R\hat{Q}'R^t}_{\bar{Q}'})^{-1} y \le 1\}
\end{aligned}
$$

9 The Ellipsoid Algorithm

Hence,

$$\bar{Q}'$$

9 The Ellipsoid Algorithm

Hence,

$$\bar{Q}' = R\hat{Q}'R^t$$

9 The Ellipsoid Algorithm

Hence,

$$\begin{aligned}\bar{Q}' &= R\hat{Q}'R^t \\ &= R \cdot \frac{n^2}{n^2-1}\Big(I - \frac{2}{n+1}e_1e_1^t\Big) \cdot R^t\end{aligned}$$

9 The Ellipsoid Algorithm

Hence,

$$\begin{aligned}\bar{Q}' &= R\hat{Q}'R^t \\ &= R \cdot \frac{n^2}{n^2-1}\Big(I - \frac{2}{n+1}e_1e_1^t\Big) \cdot R^t \\ &= \frac{n^2}{n^2-1}\Big(R \cdot R^t - \frac{2}{n+1}(Re_1)(Re_1)^t\Big)\end{aligned}$$

9 The Ellipsoid Algorithm

Hence,

$$
\begin{aligned}
\bar{Q}' &= R\hat{Q}'R^t \\
&= R \cdot \frac{n^2}{n^2-1}\Big(I - \frac{2}{n+1}e_1e_1^t\Big) \cdot R^t \\
&= \frac{n^2}{n^2-1}\Big(R \cdot R^t - \frac{2}{n+1}(Re_1)(Re_1)^t\Big) \\
&= \frac{n^2}{n^2-1}\Big(I - \frac{2}{n+1}\frac{L^taa^tL}{\|L^ta\|^2}\Big)
\end{aligned}
$$

9 The Ellipsoid Algorithm

E'

9 The Ellipsoid Algorithm

$$E' = L(\bar{E}')$$

9 The Ellipsoid Algorithm

$$\begin{aligned} E' &= L(\bar{E}') \\ &= \{L(x) \mid x^t \bar{Q}'^{-1} x \le 1\} \end{aligned}$$

9 The Ellipsoid Algorithm

$$\begin{aligned} E' &= L(\bar{E}') \\ &= \{L(x) \mid x^t \bar{Q}'^{-1} x \leq 1\} \\ &= \{y \mid (L^{-1}y)^t \bar{Q}'^{-1} L^{-1} y \leq 1\} \end{aligned}$$

9 The Ellipsoid Algorithm

$$
\begin{aligned}
E' &= L(\bar{E}') \\
&= \{L(x) \mid x^t \bar{Q}'^{-1} x \le 1\} \\
&= \{y \mid (L^{-1}y)^t \bar{Q}'^{-1} L^{-1} y \le 1\} \\
&= \{y \mid y^t (L^t)^{-1} \bar{Q}'^{-1} L^{-1} y \le 1\}
\end{aligned}
$$

9 The Ellipsoid Algorithm

$$\begin{aligned}
E' &= L(\bar{E}') \\
&= \{L(x) \mid x^t \bar{Q}'^{-1} x \le 1\} \\
&= \{y \mid (L^{-1}y)^t \bar{Q}'^{-1} L^{-1} y \le 1\} \\
&= \{y \mid y^t (L^t)^{-1} \bar{Q}'^{-1} L^{-1} y \le 1\} \\
&= \{y \mid y^t (\underbrace{L\bar{Q}'L^t}_{Q'})^{-1} y \le 1\}
\end{aligned}$$

9 The Ellipsoid Algorithm

Hence,

$$Q'$$

9 The Ellipsoid Algorithm

Hence,

$$Q' = L\bar{Q}'L^t$$

9 The Ellipsoid Algorithm

Hence,

$$\begin{aligned} Q' &= L\bar{Q}'L^t \\ &= L \cdot \frac{n^2}{n^2-1}\Big(I - \frac{2}{n+1}\frac{L^t a a^t L}{a^t Q a}\Big) \cdot L^t \end{aligned}$$

9 The Ellipsoid Algorithm

Hence,

$$\begin{aligned} Q' &= L\bar{Q}'L^t \\ &= L \cdot \frac{n^2}{n^2-1}\Bigl(I - \frac{2}{n+1}\frac{L^t a a^t L}{a^t Q a}\Bigr) \cdot L^t \\ &= \frac{n^2}{n^2-1}\Bigl(Q - \frac{2}{n+1}\frac{Q a a^t Q}{a^t Q a}\Bigr) \end{aligned}$$

Incomplete Algorithm

Algorithm 1 ellipsoid-algorithm

```
input: point $c \in \mathbb{R}^n$, convex set $K \subseteq \mathbb{R}^n$
output: point $x \in K$ or "$K$ is empty"
$Q \leftarrow$ ???
repeat
    if $c \in K$ then return $c$
    else
        choose a violated hyperplane $a$
        $c \leftarrow c - \frac{1}{n+1} \frac{Qa}{\sqrt{a^t Q a}}$
        $Q \leftarrow \frac{n^2}{n^2-1} \Big( Q - \frac{2}{n+1} \frac{Q a a^t Q}{a^t Q a} \Big)$
    endif
until ???
return "$K$ is empty"
```

Repeat: Size of basic solutions

Lemma 7

Let $P = \{x \in \mathbb{R}^n \mid Ax \le b\}$ be a bounded polytop. Let $\langle a_{\max} \rangle$ be the maximum encoding length of an entry in A. Then every entry x_j in a basic solution fulfills $|x_j| = \frac{D_j}{D}$ with $D_j, D \le 2^{2n\langle a_{\max}\rangle + n\log_2 n}$.

In the following we use $\delta := 2^{n\langle a_{\max}\rangle + n\log_2 n}$.

Note that here we have $P = \{x \mid Ax \le b\}$. The previous lemmas we had about the size of feasible solutions were slightly different as they were for different polytopes.

Repeat: Size of basic solutions

Lemma 7

Let $P = \{x \in \mathbb{R}^n \mid Ax \le b\}$ be a bounded polytop. Let $\langle a_{\max}\rangle$ be the maximum encoding length of an entry in A. Then every entry x_j in a basic solution fulfills $|x_j| = \frac{D_j}{D}$ with $D_j, D \le 2^{2n\langle a_{\max}\rangle + n\log_2 n}$.

In the following we use $\delta := 2^{n\langle a_{\max}\rangle + n\log_2 n}$.

Note that here we have $P = \{x \mid Ax \le b\}$. The previous lemmas we had about the size of feasible solutions were slightly different as they were for different polytopes.

Repeat: Size of basic solutions

Proof:

Let $\bar{A} = \begin{bmatrix} A & \\ -A & I_m \end{bmatrix}$, $\bar{b} = \begin{pmatrix} b \\ -b \end{pmatrix}$, be the matrix and right-hand vector after transforming the system to standard form.

The determinant of the matrices $\bar{A}_B$ and $\bar{M}_j$ (matrix obt. when replacing the j-th column of $\bar{A}_B$ by $\bar{b}$) can become at most

$$\begin{aligned} \det(\bar{A}_B), \det(\bar{M}_j) &\leq \|\vec{\ell}_{\max}\|^n \\ &\leq (\sqrt{n} \cdot 2^{\langle a_{\max} \rangle})^n \leq 2^{n\langle a_{\max} \rangle + n \log_2 n} \quad , \end{aligned}$$

where $\vec{\ell}_{\max}$ is the longest column-vector that can be obtained after deleting all but n rows and columns from $\bar{A}$.

This holds because columns from I_m selected when going from $\bar{A}$ to $\bar{A}_B$ do not increase the determinant. Only the at most n columns from matrices A and $-A$ that $\bar{A}$ consists of contribute.

How do we find the first ellipsoid?

For feasibility checking we can assume that the polytop P is bounded.

In this case every entry x_i in a basic solution fulfills $|x_i| \le \delta$.

Hence, P is contained in the cube $-\delta \le x_i \le \delta$.

A vector in this cube has at most distance $R := \sqrt{n}\delta$ from the origin.

Starting with the ball $E_0 := B(0, R)$ ensures that P is completely contained in the initial ellipsoid. This ellipsoid has volume at most $R^n B(0,1) \le (n\delta)^n B(0,1)$.

How do we find the first ellipsoid?

For feasibility checking we can assume that the polytop P is bounded.

In this case every entry x_i in a basic solution fulfills $|x_i| \leq \delta$.

Hence, P is contained in the cube $-\delta \leq x_i \leq \delta$.

A vector in this cube has at most distance $R := \sqrt{n}\delta$ from the origin.

Starting with the ball $E_0 := B(0, R)$ ensures that P is completely contained in the initial ellipsoid. This ellipsoid has volume at most $R^n B(0,1) \leq (n\delta)^n B(0,1)$.

How do we find the first ellipsoid?

For feasibility checking we can assume that the polytop P is bounded.

In this case every entry x_i in a basic solution fulfills $|x_i| \le \delta$.

Hence, P is contained in the cube $-\delta \le x_i \le \delta$.

A vector in this cube has at most distance $R := \sqrt{n}\delta$ from the origin.

Starting with the ball $E_0 := B(0, R)$ ensures that P is completely contained in the initial ellipsoid. This ellipsoid has volume at most $R^n B(0,1) \le (n\delta)^n B(0,1)$.

How do we find the first ellipsoid?

For feasibility checking we can assume that the polytop P is bounded.

In this case every entry x_i in a basic solution fulfills $|x_i| \le \delta$.

Hence, P is contained in the cube $-\delta \le x_i \le \delta$.

A vector in this cube has at most distance $R := \sqrt{n}\delta$ from the origin.

Starting with the ball $E_0 := B(0, R)$ ensures that P is completely contained in the initial ellipsoid. This ellipsoid has volume at most $R^n B(0,1) \le (n\delta)^n B(0,1)$.

How do we find the first ellipsoid?

For feasibility checking we can assume that the polytop P is bounded.

In this case every entry x_i in a basic solution fulfills $|x_i| \leq \delta$.

Hence, P is contained in the cube $-\delta \leq x_i \leq \delta$.

A vector in this cube has at most distance $R := \sqrt{n}\delta$ from the origin.

Starting with the ball $E_0 := B(0, R)$ ensures that P is completely contained in the initial ellipsoid. This ellipsoid has volume at most $R^n B(0,1) \leq (n\delta)^n B(0,1)$.

How do we find the first ellipsoid?

For feasibility checking we can assume that the polytop P is bounded.

In this case every entry x_i in a basic solution fulfills $|x_i| \leq \delta$.

Hence, P is contained in the cube $-\delta \leq x_i \leq \delta$.

A vector in this cube has at most distance $R := \sqrt{n}\delta$ from the origin.

Starting with the ball $E_0 := B(0, R)$ ensures that P is completely contained in the initial ellipsoid. This ellipsoid has volume at most $R^n B(0,1) \leq (n\delta)^n B(0,1)$.

When can we terminate?

Let $P := \{x \mid Ax \le b\}$ with $A \in \mathbb{Z}$ and $b \in \mathbb{Z}$ be a bounded polytop. Let $\langle a_{\max}\rangle$ be the encoding length of the largest entry in A or b.

Consider the following polytope

$$P_\lambda := \left\{x \mid Ax \le b + \frac{1}{\lambda}\begin{pmatrix}1\\ \vdots\\ 1\end{pmatrix}\right\} ,$$

where $\lambda = \delta^2 + 1$.

When can we terminate?

Let $P := \{x \mid Ax \le b\}$ with $A \in \mathbb{Z}$ and $b \in \mathbb{Z}$ be a bounded polytop. Let $\langle a_{\max} \rangle$ be the encoding length of the largest entry in A or b.

Consider the following polytope

$$P_\lambda := \left\{ x \mid Ax \le b + \frac{1}{\lambda} \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} \right\} ,$$

where $\lambda = \delta^2 + 1$.

When can we terminate?

Let $P := \{x \mid Ax \le b\}$ with $A \in \mathbb{Z}$ and $b \in \mathbb{Z}$ be a bounded polytop. Let $\langle a_{\max}\rangle$ be the encoding length of the largest entry in A or b.

Consider the following polytope

$$P_\lambda := \left\{ x \mid Ax \le b + \frac{1}{\lambda}\begin{pmatrix}1\\ \vdots\\ 1\end{pmatrix} \right\} ,$$

where $\lambda = \delta^2 + 1$.

Lemma 8

P_λ is feasible if and only if P is feasible.

$\Longleftarrow$: obvious!

Lemma 8

P_λ is feasible if and only if P is feasible.

$\Leftarrow$: obvious!

$\Longrightarrow$:

Consider the polytops

$$\bar{P} = \left\{ x \mid \left[\begin{matrix} A \\ -A \end{matrix} \; I_m \right] x = \begin{pmatrix} b \\ -b \end{pmatrix} ; x \geq 0 \right\}$$

and

$$\bar{P}_\lambda = \left\{ x \mid \left[\begin{matrix} A \\ -A \end{matrix} \; I_m \right] x = \begin{pmatrix} b \\ -b \end{pmatrix} + \frac{1}{\lambda} \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} ; x \geq 0 \right\} \; .$$

P is feasible if and only if $\bar{P}$ is feasible, and P_λ feasible if and only if $\bar{P}_\lambda$ feasible.

$\bar{P}_\lambda$ is bounded since P_λ and P are bounded.

$\Longrightarrow$:

Consider the polytops

$$\bar{P} = \left\{ x \mid \begin{bmatrix} A & \\ -A & I_m \end{bmatrix} x = \begin{pmatrix} b \\ -b \end{pmatrix}; x \geq 0 \right\}$$

and

$$\bar{P}_\lambda = \left\{ x \mid \begin{bmatrix} A & \\ -A & I_m \end{bmatrix} x = \begin{pmatrix} b \\ -b \end{pmatrix} + \frac{1}{\lambda} \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}; x \geq 0 \right\} .$$

P is feasible if and only if $\bar{P}$ is feasible, and P_λ feasible if and only if $\bar{P}_\lambda$ feasible.

$\bar{P}_\lambda$ is bounded since P_λ and P are bounded.

⟹:

Consider the polytops

$$\bar{P} = \left\{x \mid \left[\begin{matrix} A \\ -A \end{matrix}\; I_m\right] x = \begin{pmatrix} b \\ -b \end{pmatrix}; x \geq 0\right\}$$

and

$$\bar{P}_\lambda = \left\{x \mid \left[\begin{matrix} A \\ -A \end{matrix}\; I_m\right] x = \begin{pmatrix} b \\ -b \end{pmatrix} + \frac{1}{\lambda}\begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}; x \geq 0\right\} \; .$$

P is feasible if and only if $\bar{P}$ is feasible, and P_λ feasible if and only if $\bar{P}_\lambda$ feasible.

$\bar{P}_\lambda$ is bounded since P_λ and P are bounded.

$\Longrightarrow$:

Consider the polytops

$$\bar{P} = \left\{x \mid \begin{bmatrix} A \\ -A \end{bmatrix} I_m \Big] x = \begin{pmatrix} b \\ -b \end{pmatrix}; x \geq 0\right\}$$

and

$$\bar{P}_\lambda = \left\{x \mid \begin{bmatrix} A \\ -A \end{bmatrix} I_m \Big] x = \begin{pmatrix} b \\ -b \end{pmatrix} + \frac{1}{\lambda}\begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}; x \geq 0\right\} .$$

P is feasible if and only if $\bar{P}$ is feasible, and P_λ feasible if and only if $\bar{P}_\lambda$ feasible.

$\bar{P}_\lambda$ is bounded since P_λ and P are bounded.

Let $\bar{A} = \begin{bmatrix} \begin{matrix} A \\ -A \end{matrix} & I_m \end{bmatrix}$, and $\bar{b} = \begin{pmatrix} b \\ -b \end{pmatrix}$.

$\bar{P}_\lambda$ feasible implies that there is a basic feasible solution represented by

$$x_B = \bar{A}_B^{-1}\bar{b} + \frac{1}{\lambda}\bar{A}_B^{-1}\begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$$

(The other x-values are zero)

The only reason that this basic feasible solution is not feasible for $\bar{P}$ is that one of the basic variables becomes negative.

Hence, there exists i with

$$(\bar{A}_B^{-1}\bar{b})_i < 0 \le (\bar{A}_B^{-1}\bar{b})_i + \frac{1}{\lambda}(\bar{A}_B^{-1}\vec{1})_i$$

Let $\bar{A} = \begin{bmatrix} A & \\ -A & I_m \end{bmatrix}$, and $\bar{b} = \begin{pmatrix} b \\ -b \end{pmatrix}$.

$\bar{P}_\lambda$ feasible implies that there is a basic feasible solution represented by

$$x_B = \bar{A}_B^{-1}\bar{b} + \frac{1}{\lambda}\bar{A}_B^{-1}\begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$$

(The other x-values are zero)

The only reason that this basic feasible solution is not feasible for $\bar{P}$ is that one of the basic variables becomes negative.

Hence, there exists i with

$$(\bar{A}_B^{-1}\bar{b})_i < 0 \le (\bar{A}_B^{-1}\bar{b})_i + \frac{1}{\lambda}(\bar{A}_B^{-1}\vec{1})_i$$

Let $\bar{A} = \begin{bmatrix} A & \\ -A & I_m \end{bmatrix}$, and $\bar{b} = \begin{pmatrix} b \\ -b \end{pmatrix}$.

$\bar{P}_\lambda$ feasible implies that there is a basic feasible solution represented by

$$x_B = \bar{A}_B^{-1}\bar{b} + \frac{1}{\lambda}\bar{A}_B^{-1}\begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$$

(The other x-values are zero)

The only reason that this basic feasible solution is not feasible for $\bar{P}$ is that one of the basic variables becomes negative.

Hence, there exists i with

$$(\bar{A}_B^{-1}\bar{b})_i < 0 \le (\bar{A}_B^{-1}\bar{b})_i + \frac{1}{\lambda}(\bar{A}_B^{-1}\vec{1})_i$$

By Cramers rule we get

$$(\bar{A}_B^{-1}\bar{b})_i < 0 \quad \Longrightarrow \quad (\bar{A}_B^{-1}\bar{b})_i \le -\frac{1}{\det(\bar{A}_B)}$$

and

$$(\bar{A}_B^{-1}\vec{1})_i \le \det(\bar{M}_j) \ \ ,$$

where $\bar{M}_j$ is obtained by replacing the j-th column of $\bar{A}_B$ by $\vec{1}$.

However, we showed that the determinants of $\bar{A}_B$ and $\bar{M}_j$ can become at most δ.

Since, we chose $\lambda = \delta^2 + 1$ this gives a contradiction.

By Cramers rule we get

$$(\bar{A}_B^{-1}\bar{b})_i < 0 \quad \Longrightarrow \quad (\bar{A}_B^{-1}\bar{b})_i \le -\frac{1}{\det(\bar{A}_B)}$$

and

$$(\bar{A}_B^{-1}\vec{1})_i \le \det(\bar{M}_j) \enspace ,$$

where $\bar{M}_j$ is obtained by replacing the j-th column of $\bar{A}_B$ by $\vec{1}$.

However, we showed that the determinants of $\bar{A}_B$ and $\bar{M}_j$ can become at most δ.

Since, we chose $\lambda = \delta^2 + 1$ this gives a contradiction.

By Cramers rule we get

$$(\bar{A}_B^{-1}\bar{b})_i < 0 \quad \Longrightarrow \quad (\bar{A}_B^{-1}\bar{b})_i \leq -\frac{1}{\det(\bar{A}_B)}$$

and

$$(\bar{A}_B^{-1}\vec{1})_i \leq \det(\bar{M}_j) \ ,$$

where $\bar{M}_j$ is obtained by replacing the j-th column of $\bar{A}_B$ by $\vec{1}$.

However, we showed that the determinants of $\bar{A}_B$ and $\bar{M}_j$ can become at most δ.

Since, we chose $\lambda = \delta^2 + 1$ this gives a contradiction.

Lemma 9

If P_λ is feasible then it contains a ball of radius $r := 1/\delta^3$. This has a volume of at least $r^n\mathrm{vol}(B(0,1) = \frac{1}{\delta^{3n}}\mathrm{vol}(B(0,1))$.

Lemma 9

If P_λ is feasible then it contains a ball of radius $r := 1/\delta^3$. This has a volume of at least $r^n \mathrm{vol}(B(0,1) = \frac{1}{\delta^{3n}} \mathrm{vol}(B(0,1))$.

Proof:

If P_λ feasible then also P. Let x be feasible for P.

Lemma 9

If P_λ is feasible then it contains a ball of radius $r := 1/\delta^3$. This has a volume of at least $r^n \mathrm{vol}(B(0,1) = \frac{1}{\delta^{3n}} \mathrm{vol}(B(0,1))$.

Proof:

If P_λ feasible then also P. Let x be feasible for P.

This means $Ax \le b$.

Lemma 9

If P_λ is feasible then it contains a ball of radius $r := 1/\delta^3$. This has a volume of at least $r^n \mathrm{vol}(B(0,1) = \frac{1}{\delta^{3n}}\mathrm{vol}(B(0,1))$.

Proof:

If P_λ feasible then also P. Let x be feasible for P.
This means $Ax \leq b$.

Let $\vec{\ell}$ with $\|\vec{\ell}\| \leq r$. Then

$$(A(x + \vec{\ell}))_i$$

Lemma 9

If P_λ is feasible then it contains a ball of radius $r := 1/\delta^3$. This has a volume of at least $r^n \mathrm{vol}(B(0,1) = \frac{1}{\delta^{3n}} \mathrm{vol}(B(0,1))$.

Proof:

If P_λ feasible then also P. Let x be feasible for P.
This means $Ax \le b$.

Let $\vec{\ell}$ with $\|\vec{\ell}\| \le r$. Then

$$(A(x + \vec{\ell}))_i = (Ax)_i + (A\vec{\ell})_i$$

Lemma 9

If P_λ is feasible then it contains a ball of radius $r := 1/\delta^3$. This has a volume of at least $r^n \mathrm{vol}(B(0,1)) = \frac{1}{\delta^{3n}} \mathrm{vol}(B(0,1))$.

Proof:

If P_λ feasible then also P. Let x be feasible for P.
This means $Ax \le b$.

Let $\vec{\ell}$ with $\|\vec{\ell}\| \le r$. Then

$$(A(x+\vec{\ell}))_i = (Ax)_i + (A\vec{\ell})_i \le b_i + A_i\vec{\ell}$$

Lemma 9

If P_λ is feasible then it contains a ball of radius $r := 1/\delta^3$. This has a volume of at least $r^n \mathrm{vol}(B(0,1) = \frac{1}{\delta^{3n}} \mathrm{vol}(B(0,1))$.

Proof:

If P_λ feasible then also P. Let x be feasible for P.
This means $Ax \le b$.

Let $\vec{\ell}$ with $\|\vec{\ell}\| \le r$. Then

$$\begin{aligned}(A(x + \vec{\ell}))_i &= (Ax)_i + (A\vec{\ell})_i \le b_i + A_i\vec{\ell} \\ &\le b_i + \|A_i\| \cdot \|\vec{\ell}\|\end{aligned}$$

Lemma 9

If P_λ is feasible then it contains a ball of radius $r := 1/\delta^3$. This has a volume of at least $r^n \mathrm{vol}(B(0,1) = \frac{1}{\delta^{3n}}\mathrm{vol}(B(0,1))$.

Proof:

If P_λ feasible then also P. Let x be feasible for P.
This means $Ax \le b$.

Let $\vec{\ell}$ with $\|\vec{\ell}\| \le r$. Then

$$\begin{aligned}(A(x + \vec{\ell}))_i &= (Ax)_i + (A\vec{\ell})_i \le b_i + A_i\vec{\ell} \\ &\le b_i + \|A_i\| \cdot \|\vec{\ell}\| \le b_i + \sqrt{n} \cdot 2^{\langle a_{\max} \rangle} \cdot r\end{aligned}$$

Lemma 9

If P_λ is feasible then it contains a ball of radius $r := 1/\delta^3$. This has a volume of at least $r^n \mathrm{vol}(B(0,1) = \frac{1}{\delta^{3n}}\mathrm{vol}(B(0,1))$.

Proof:

If P_λ feasible then also P. Let x be feasible for P.
This means $Ax \leq b$.

Let $\vec{\ell}$ with $\|\vec{\ell}\| \leq r$. Then

$$\begin{aligned}(A(x+\vec{\ell}))_i &= (Ax)_i + (A\vec{\ell})_i \leq b_i + A_i\vec{\ell} \\ &\leq b_i + \|A_i\| \cdot \|\vec{\ell}\| \leq b_i + \sqrt{n} \cdot 2^{\langle a_{\max} \rangle} \cdot r \\ &\leq b_i + \frac{\sqrt{n} \cdot 2^{\langle a_{\max} \rangle}}{\delta^3}\end{aligned}$$

Lemma 9

If P_λ is feasible then it contains a ball of radius $r := 1/\delta^3$. This has a volume of at least $r^n \mathrm{vol}(B(0,1) = \frac{1}{\delta^{3n}}\mathrm{vol}(B(0,1))$.

Proof:

If P_λ feasible then also P. Let x be feasible for P.
This means $Ax \le b$.

Let $\vec{\ell}$ with $\|\vec{\ell}\| \le r$. Then

$$\begin{aligned}(A(x+\vec{\ell}))_i &= (Ax)_i + (A\vec{\ell})_i \le b_i + A_i\vec{\ell}\\ &\le b_i + \|A_i\| \cdot \|\vec{\ell}\| \le b_i + \sqrt{n}\cdot 2^{\langle a_{\max}\rangle}\cdot r\\ &\le b_i + \frac{\sqrt{n}\cdot 2^{\langle a_{\max}\rangle}}{\delta^3} \le b_i + \frac{1}{\delta^2+1} \le b_i + \frac{1}{\lambda}\end{aligned}$$

Lemma 9

If P_λ is feasible then it contains a ball of radius $r := 1/\delta^3$. This has a volume of at least $r^n \mathrm{vol}(B(0,1)) = \frac{1}{\delta^{3n}}\mathrm{vol}(B(0,1))$.

Proof:

If P_λ feasible then also P. Let x be feasible for P.
This means $Ax \leq b$.

Let $\vec{\ell}$ with $\|\vec{\ell}\| \leq r$. Then

$$\begin{aligned}(A(x+\vec{\ell}))_i &= (Ax)_i + (A\vec{\ell})_i \leq b_i + A_i\vec{\ell}\\ &\leq b_i + \|A_i\| \cdot \|\vec{\ell}\| \leq b_i + \sqrt{n} \cdot 2^{\langle a_{\max}\rangle} \cdot r\\ &\leq b_i + \frac{\sqrt{n} \cdot 2^{\langle a_{\max}\rangle}}{\delta^3} \leq b_i + \frac{1}{\delta^2+1} \leq b_i + \frac{1}{\lambda}\end{aligned}$$

Hence, $x + \vec{\ell}$ is feasible for P_λ which proves the lemma.

How many iterations do we need until the volume becomes too small?

How many iterations do we need until the volume becomes too small?

$$e^{-\frac{i}{2(n+1)}} \cdot \mathrm{vol}(B(0,R)) < \mathrm{vol}(B(0,r))$$

How many iterations do we need until the volume becomes too small?

$$e^{-\frac{i}{2(n+1)}} \cdot \mathrm{vol}(B(0,R)) < \mathrm{vol}(B(0,r))$$

Hence,

$$i$$

How many iterations do we need until the volume becomes too small?

$$e^{-\frac{i}{2(n+1)}} \cdot \mathrm{vol}(B(0,R)) < \mathrm{vol}(B(0,r))$$

Hence,

$$i > 2(n+1)\ln\left(\frac{\mathrm{vol}(B(0,R))}{\mathrm{vol}(B(0,r))}\right)$$

How many iterations do we need until the volume becomes too small?

$$e^{-\frac{i}{2(n+1)}} \cdot \mathrm{vol}(B(0,R)) < \mathrm{vol}(B(0,r))$$

Hence,

$$\begin{aligned} i &> 2(n+1)\ln\left(\frac{\mathrm{vol}(B(0,R))}{\mathrm{vol}(B(0,r))}\right) \\ &= 2(n+1)\ln\left(n^n\delta^n \cdot \delta^{3n}\right) \end{aligned}$$

How many iterations do we need until the volume becomes too small?

$$e^{-\frac{i}{2(n+1)}} \cdot \mathrm{vol}(B(0,R)) < \mathrm{vol}(B(0,r))$$

Hence,

$$\begin{aligned} i &> 2(n+1)\ln\left(\frac{\mathrm{vol}(B(0,R))}{\mathrm{vol}(B(0,r))}\right) \\ &= 2(n+1)\ln\left(n^n\delta^n \cdot \delta^{3n}\right) \\ &= 8n(n+1)\ln(\delta) + 2(n+1)n\ln(n) \end{aligned}$$

How many iterations do we need until the volume becomes too small?

$$e^{-\frac{i}{2(n+1)}} \cdot \mathrm{vol}(B(0,R)) < \mathrm{vol}(B(0,r))$$

Hence,

$$\begin{aligned}
i &> 2(n+1)\ln\left(\frac{\mathrm{vol}(B(0,R))}{\mathrm{vol}(B(0,r))}\right) \\
&= 2(n+1)\ln\left(n^n\delta^n \cdot \delta^{3n}\right) \\
&= 8n(n+1)\ln(\delta) + 2(n+1)n\ln(n) \\
&= \mathcal{O}(\mathrm{poly}(n, \langle a_{\max}\rangle))
\end{aligned}$$

Algorithm 1 ellipsoid-algorithm

```
input: point $c \in \mathbb{R}^n$, convex set $K \subseteq \mathbb{R}^n$, radii $R$ and $r$
        with $K \subseteq B(0,R)$, and $B(x,r) \subseteq K$ for some $x$
output: point $x \in K$ or "$K$ is empty"
$Q \leftarrow \mathrm{diag}(R^2,\ldots,R^2)$ // i.e., $L = \mathrm{diag}(R,\ldots,R)$
$c \leftarrow 0$
repeat
    if $c \in K$ then return $c$
    else
        choose a violated hyperplane $a$
        $c \leftarrow c - \frac{1}{n+1}\frac{Qa}{\sqrt{a^tQa}}$
        $Q \leftarrow \frac{n^2}{n^2-1}\Big(Q - \frac{2}{n+1}\frac{Qaa^tQ}{a^tQa}\Big)$
    endif
until $\det(Q) \le r^{2n}$ // i.e., $\det(L) \le r^n$
return "$K$ is empty"
```

Separation Oracle:

Let $K \subseteq \mathbb{R}^n$ be a convex set. A separation oracle for K is an algorithm A that gets as input a point $x \in \mathbb{R}^n$ and either

- certifies that $x \in K$,
- or finds a hyperplane separating x from K.

We will usually assume that A is a polynomial-time algorithm.

In order to find a point in K we need

- a guarantee that a ball of radius r is contained in K,
- an initial ball $B(c, R)$ with radius R that contains K,
- a separation oracle for K.

The Ellipsoid algorithm requires $\mathcal{O}(\text{poly}(n) \cdot \log(R/r))$ iterations. Each iteration is polytime for a polynomial-time Separation oracle.

Separation Oracle:
Let $K \subseteq \mathbb{R}^n$ be a convex set. A separation oracle for K is an algorithm A that gets as input a point $x \in \mathbb{R}^n$ and either

- certifies that $x \in K$,
- or finds a hyperplane separating x from K.

We will usually assume that A is a polynomial-time algorithm.

In order to find a point in K we need

- a guarantee that a ball of radius r is contained in K,
- an initial ball $B(c, R)$ with radius R that contains K,
- a separation oracle for K.

The Ellipsoid algorithm requires $\mathcal{O}(\text{poly}(n) \cdot \log(R/r))$ iterations. Each iteration is polytime for a polynomial-time Separation oracle.

Separation Oracle:

Let $K \subseteq \mathbb{R}^n$ be a convex set. A separation oracle for K is an algorithm A that gets as input a point $x \in \mathbb{R}^n$ and either

- certifies that $x \in K$,
- or finds a hyperplane separating x from K.

We will usually assume that A is a polynomial-time algorithm.

In order to find a point in K we need

- a guarantee that a ball of radius r is contained in K,
- an initial ball $B(c, R)$ with radius R that contains K,
- a separation oracle for K.

The Ellipsoid algorithm requires $\mathcal{O}(\mathrm{poly}(n) \cdot \log(R/r))$ iterations. Each iteration is polytime for a polynomial-time Separation oracle.

Separation Oracle:

Let $K \subseteq \mathbb{R}^n$ be a convex set. A separation oracle for K is an algorithm A that gets as input a point $x \in \mathbb{R}^n$ and either

- certifies that $x \in K$,
- or finds a hyperplane separating x from K.

We will usually assume that A is a polynomial-time algorithm.

In order to find a point in K we need

- a guarantee that a ball of radius r is contained in K,
- an initial ball $B(c, R)$ with radius R that contains K,
- a separation oracle for K.

The Ellipsoid algorithm requires $\mathcal{O}(\text{poly}(n) \cdot \log(R/r))$ iterations. Each iteration is polytime for a polynomial-time Separation oracle.

Separation Oracle:

Let $K \subseteq \mathbb{R}^n$ be a convex set. A separation oracle for K is an algorithm A that gets as input a point $x \in \mathbb{R}^n$ and either

- certifies that $x \in K$,
- or finds a hyperplane separating x from K.

We will usually assume that A is a polynomial-time algorithm.

In order to find a point in K we need

- a guarantee that a ball of radius r is contained in K,
- an initial ball $B(c, R)$ with radius R that contains K,
- a separation oracle for K.

The Ellipsoid algorithm requires $\mathcal{O}(\mathrm{poly}(n) \cdot \log(R/r))$ iterations. Each iteration is polytime for a polynomial-time Separation oracle.

Separation Oracle:

Let $K \subseteq \mathbb{R}^n$ be a convex set. A separation oracle for K is an algorithm A that gets as input a point $x \in \mathbb{R}^n$ and either

- certifies that $x \in K$,
- or finds a hyperplane separating x from K.

We will usually assume that A is a polynomial-time algorithm.

In order to find a point in K we need

- a guarantee that a ball of radius r is contained in K,
- an initial ball $B(c, R)$ with radius R that contains K,
- a separation oracle for K.

The Ellipsoid algorithm requires $\mathcal{O}(\text{poly}(n) \cdot \log(R/r))$ iterations. Each iteration is polytime for a polynomial-time Separation oracle.

Separation Oracle:

Let $K \subseteq \mathbb{R}^n$ be a convex set. A separation oracle for K is an algorithm A that gets as input a point $x \in \mathbb{R}^n$ and either

- certifies that $x \in K$,
- or finds a hyperplane separating x from K.

We will usually assume that A is a polynomial-time algorithm.

In order to find a point in K we need

- a guarantee that a ball of radius r is contained in K,
- an initial ball $B(c, R)$ with radius R that contains K,
- a separation oracle for K.

The Ellipsoid algorithm requires $\mathcal{O}(\text{poly}(n) \cdot \log(R/r))$ iterations. Each iteration is polytime for a polynomial-time Separation oracle.