von Neumann's Algorithm for feasibility:

To determine if $c \in CH(a_1, \dots, a_m)$

and to find y s.t. $Ay \approx c$

Is easy if $m = d+1$ and rank $\begin{pmatrix} a_1 & & a_{d+1} \\ 1 & , \dots, & 1 \end{pmatrix} = d+1$:

just solve $\begin{pmatrix} A \\ 1^T \end{pmatrix} y = \begin{pmatrix} c \\ 1 \end{pmatrix}$ and check if y is ≥ 0.

Difficulty comes from having $m > d+1$ vectors.

Unlike simplex, this is iterative, with more iterations giving better approximations.

Simplification 1: Is equivalent to determine if $0 \in CH(\hat{a}_1, \dots \hat{a}_m)$, where $\hat{a}_i = a_i - c$.

<u>proof</u> $1^T y = 1 \Rightarrow \sum y(i)\hat{a}_i = \bar{0} \Leftrightarrow \sum y(i) a_i = c$

Simplification 2: Is equivalent to determine if $0 \in CH(a_1/\|a_1\|, \dots, a_m/\|a_m\|)$, but this is special for 0.

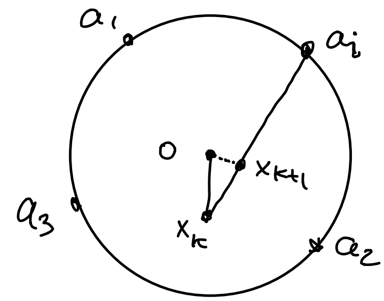
<u>proof</u> $0 \notin CH(a_1 \dots a_m)$ iff $\exists$ sep. hyperplane iff $\exists\ x$ s.t. $x^T a_i > 0$, $\forall i$.

And, this does not depend on norm of a_i.

Real goal: find $Ay \approx c$ or find a separating hyperplane

To start, let $y_0 = \frac{1}{m}\mathbb{1}$, $x_0 = Ay_0$, a convex combination.
At step k, construct $x_k = Ay_k$, and measure $\|x_k\|_2$

Motivating picture:



At step $k+1$, choose i to maximize $x_k^T(x_k - a_i)$

If we view $-x_k$ as the error to be corrected,

then think of maximizing $(-x_k)^T(a_i - x_k)$

This minimizes the angle.

If $x_k^T(x_k - a_i) < 0$, $\forall i$, then we have found a separating hyperplane: $0 \le x_k^T x_k < x_k^T a_i$, $\forall i$

(Same if $x_k^T a_i > 0$, $\forall i$)

If $x_k^T(x_k - a_i) \ge 0$ we will set x_{k+1} to be the point of least norm on the segment $\overline{x_k a_i}$.

These points are in the convex hull because they have the form

$$(1-\lambda)x_k + \lambda a_i = A\left((1-\lambda)y_k + \lambda e_i\right), \quad 0 \le \lambda \le 1$$

Writing $(1-\lambda)x_k + \lambda a_i = x_k + \lambda(a_i - x_k)$,

we compute the squared norm to be

$$\|x_k\|_2^2 + \lambda^2\|a_i - x_k\|_2^2 + 2\lambda x_k^T(a_i - x_k).$$

The optimal value of λ is $\frac{x_k^T(x_k - a_i)}{\|x_k - a_i\|_2^2}$,

and using this to choose x_{k+1} gives

$$\|x_{k+1}\|^2 = \|x_k\|^2 - \frac{(x_k^T(x_k - a_i))^2}{\|x_k - a_i\|_2^2} \qquad (*$$

Thm 1 (Dantzig)

If $0 \in CH(a_1, \dots, a_m)$, then $\|x_k\| \le 2/\sqrt{k}$.

proof

As $a_i \in B(0,1)$ and $x_k \in CH(a_1, \dots, a_m)$, $x_k \in B(0,1)$

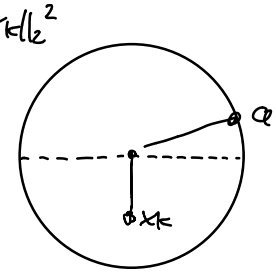
$\iff \|x_k\|_2 \le 1$.

So, the statement is trivially true for $k \le 4$.

We prove it by induction.

As $0 \in CH(a_1 \dots a_m)$, $\exists j$ s.t. $x_k^T a_j \le 0$

for this j, $x_k^T(x_k - a_j) \ge x_k^T x_k = \|x_k\|_2^2$



So, for the chosen i,

$x_k^T(x_k - a_i) \ge \|x_k\|_2^2$

As $\|x_k\|_2 \le 1$ and $\|a_i\|_2 \le 1$, $\|x_k - a_i\|_2^2 \le 4$

Combining gives $\dfrac{\left(x_k^T(x_k - a_i)\right)^2}{\|x_k - a_i\|_2^2} \geq \dfrac{\|x_k\|_2^4}{4}$

and $\|x_{k+1}\|_2^2 \leq \|x_k\|_2^2 - \|x_k\|_2^2 \dfrac{\|x_k\|_2^2}{4}$

$$= \|x_k\|_2^2\left(1 - \frac{\|x_k\|_2^2}{4}\right)$$

Let $f(z) = z\left(1 - \frac{z}{4}\right)$

$f'(z) = 1 - z/2 \quad \leq 0$ for $z \geq 2$, so

this is monotonically decreasing for $z \geq 2$

Thus, it suffices to show $f(4/k) \leq 4/(k+1)$ for $k \geq 2$

This follows from

$$\frac{4}{k}\left(1 - \frac{4}{4k}\right) = \frac{4}{k}\left(1 - \frac{1}{k}\right) = 4\,\frac{k-1}{k^2} \leq 4\,\frac{k-1}{k^2-1} = 4\,\frac{1}{k+1}$$

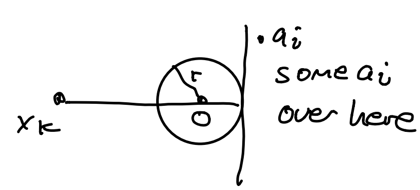
<u>Thm 2</u> (Freund & Epelman) If $B(0, r) \in CH(a_1 \dots a_m)$ (1)

$$\|x_k\|_2 \leq \exp\left(\frac{-kr^2}{8}\right)$$

<u>proof</u> We will show $\|x_{k+1}\|^2 \leq \|x_k\|^2\left(1 - \frac{r^2}{4}\right)$

Use $\|x_{k+1}\|^2 = \|x_k\|^2 - \dfrac{\left(x_k^T(x_k - a_i)\right)^2}{\|x_k - a_i\|_2^2}$

(1) $\Rightarrow \exists i$ s.t. $-\dfrac{x_k^T}{\|x_k\|} a_i \geq r$



$$\Rightarrow \frac{x_k^T}{\|x_k\|}(x_k - a_i) \geq r$$

$$\Rightarrow x_k^T(x_k - a_i) \geq r \cdot \|x_k\|$$

$$\Rightarrow \frac{(x_k^T(x_k - a_i))^2}{\|x_k - a_i\|_2^2} \geq \|x_k\|^2 \frac{r^2}{4}$$

So, $\|x_k\|_2^2 \leq \left(1 - \frac{r^2}{4}\right)^k$,

$$\|x_k\|_2 \leq \left(1 - \frac{r^2}{4}\right)^{k/2} \leq \exp\left(-\frac{r^2}{4}\right)^{k/2} = \exp\left(\frac{-kr^2}{8}\right)$$

So, get $\|x_k\| \leq \varepsilon$ by $\exp\left(\frac{-kr^2}{8}\right) \leq \varepsilon$

$$\Leftrightarrow \frac{kr^2}{8} \geq \ln(1/\varepsilon)$$

$$k \geq 8\ln(1/\varepsilon)/r^2$$

If we use this algorithm to max α s.t. $\alpha c \in CH$, it slows down as αc approaches the boundary.

Condition numbers.

Let r be maximum s.t. $B(c, r) \in CH(a_1, \ldots, a_m)$

r measures distance to infeasibility

(obvious direction is $r \leq$ dist to infeasible - move c)

Renegar proved $1/r$ is a condition number.

We say the problem is <u>ill-posed</u> when
$c \in$ boundary $(CH(a_1 .. a_m))$
(so can deal with feasible & infeasible)

Let K = dist-to-ill-posed.

That is $K = \min \|\delta_1\| + \cdots + \|\delta_m\| + \|\gamma\|$ s.t.
$c + \gamma \in \text{bdry}(CH(a_1+\delta_1, \ldots, a_m+\delta_m))$.

<u>lem</u> $K = r$.

<u>proof</u> $K \le r$, because c is dist r from bdry$(CH(a_1 .. a_m))$

To show $K \ge r$, let $c + \gamma \in \text{bdry}(CH(a_i + \delta_i)_i)$

$\exists$ a supporting hyperplane given by $\|x\| = 1$, so
$x^T(c+\gamma) \ge x^T(a_i + \delta_i) \quad \forall i$.
$\Rightarrow x^T c \ge x^T a_i + x^T \delta_i - x^T \gamma \ge x^T a_i - \|\delta_i\| - \|\gamma\|$

On the other hand, $B(c, r) \in CH(a_1 .. a_m)$
$\Rightarrow \exists i$ s.t. $x^T a_i \ge x^T c + r$

So, $x^T c \ge x^T c + r - \|\delta_i\| - \|\gamma\|$
$\Rightarrow \|\delta_i\| + \|\gamma\| \ge r$.

Running Times:

Each step of this algorithm requires computing $a_i^T x$ for all $i \to$ time $\approx md$ per iteration.

What about Simplex? Practical experience is that it needs $\leq O(m)$ steps. But, in each must solve a system of equations in d variables, which can take time $\sim d^3$ (or d^ω).

But, can make faster because only change one column of the matrix in which solve the system.

Sherman-Morrison

$$(M + uv^T)^{-1} = M^{-1} - \frac{M^{-1} u v^T M^{-1}}{1 + v^T M^{-1} u}$$

So, if change one column can update an inverse in $\leq O(d^2)$ time. **Is key to many fast algorithms**

Can do for LU-factorization, too.

Interior Point Methods: time $\leq O(m^3 \lg(\kappa/\varepsilon))$,

ε-accurate, condition # κ

Is logarithmic in condition #!

Most recent Feb 6, '20:

Solving Tall Dense Linear Programs
in Nearly Linear Time

Jan van den Brand* Yin Tat Lee† Aaron Sidford‡ Zhao Song§

$\tilde{O}(m + d^3)$