Geometría convexa y politopos, día 5

Alexey Beshenov (cadadr@gmail.com)

12 de agosto de 2016

9. Puntos extremos y el teorema de Krein–Milman

Sea K un conjunto convexo cerrado. Les recuerdo que $\mathbf{x} \in K$ se llama un **punto extremo** si $\{\mathbf{x}\} \subset K$ es un subconjunto extremo, que es la misma cosa que $K \setminus \{\mathbf{x}\}$ sea convexo.

9.1. Ejemplo. Aquí está un plano de estadio de fútbol con sus puntos extremos representados por las líneas gruesas.

▲

El siguiente hecho demuestra el significado de los puntos extremos:

9.2. Teorema (Krein, Milman). *Sea $K \subset \mathbb{R}^n$ un conjunto convexo y compacto. Sea*

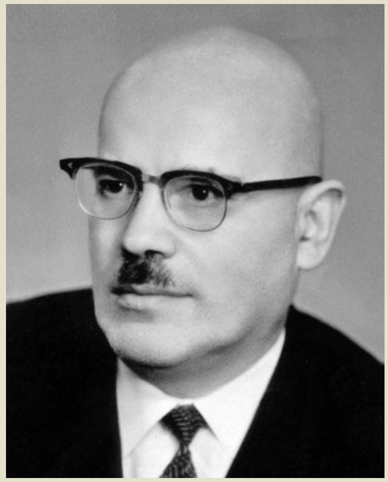
$$\operatorname{ex}(K) := \{\mathbf{x} \in K \mid \{\mathbf{x}\} \subset K \text{ es extremo}\} = \{\mathbf{x} \in K \mid K \setminus \{\mathbf{x}\} \text{ es convexo}\}.$$

el conjunto de sus puntos extremos. Entonces

$$K = \operatorname{conv} \operatorname{ex} K.$$

Demostración. Obviamente, $\operatorname{conv} \operatorname{ex} K \subseteq K$ y tenemos que demostrar que $K \subseteq \operatorname{conv} \operatorname{ex} K$. Usamos inducción sobre la dimensión. En dimensiones 0 y 1, cuando K es un punto o un segmento cerrado, está claro que $K = \operatorname{conv} \operatorname{ex} K$. Para cualquier punto $\mathbf{x} \in K$ sea ℓ una recta que pasa por $\mathbf{x}$. Tenemos $[\mathbf{x}_1, \mathbf{x}_2] := \ell \cap K$ (porque K es compacto por nuestra hipótesis) para algunos puntos $\mathbf{x}_1, \mathbf{x}_2$ en la frontera $\operatorname{fr} K$. Por convexidad de $\operatorname{conv} \operatorname{ex} K$, es suficiente demostrar que $\mathbf{x}_1, \mathbf{x}_2 \in \operatorname{conv} \operatorname{ex} K$. Para $i = 1, 2$ sea H_i un hiperplano de soporte que pasa por $\mathbf{x}_i$. El subconjunto $H_i \cap K$ tiene dimensión menor y podemos aplicar la hipótesis de inducción a $H_i \cap K$ como un subconjunto del espacio afín H_i. Entonces $H_i \cap K = \operatorname{conv} \operatorname{ex}(H_i \cap K)$. Pero $\operatorname{ex}(H_i \cap K) = H_i \cap \operatorname{ex} K \subseteq \operatorname{ex} K$, entonces $H_i \cap K \subseteq \operatorname{conv} \operatorname{ex} K$. ■

La proposición de arriba (para un subconjunto convexo compacto de $\mathbb{R}^n$) se atribuye a Minkowski, pero en su mayor generalidad se conoce como el **teorema de Krein–Milman** y fue demostrada por los matemáticos soviéticos Mark Krein (1907–1989) y su estudiante David Milman (1912–1982) que trabajaban en el análisis funcional. Krein era estudiante de Nikolái Chebotariov (1894–1947), otro matemático soviético famoso por sus resultados en teoría de números. Krein y Milman trabajaban en Odessa (Ucrania). Durante la guerra la universidad fue evacuada, y después de su regreso en 1944 Krein fue despedido por acusaciones del "nacionalismo judío" y tuvo posiciones en el Instituto de ingeniería marítima y el Instituto de ingeniería civil de Odessa. No obstante la persecución política y imposibilidad de atender los congresos en el extranjero, Krein logró reconocimiento internacional y publicó más de 270 artículos.



Mark Krein

Milman también fue discriminado por sus orígenes judíos, fue despedido y tuvo que trabajar en el Instituto de telecomunicaciones de Odessa. En 1973, después de su jubilación, se mudó a Israel, como muchos otros judíos soviéticos.

10. Teorema de Minkowski–Weyl

10.1. Teorema (Minkowski, Weyl). *Para un conjunto $P \subset \mathbb{R}^n$ las siguientes afirmaciones son equivalentes:*

1) P es un poliedro (una intersección de un número finito de semiespacios cerrados) y P es acotado,

2) P es un politopo convexo (la envolvente convexa de un conjunto finito $\{\mathbf{x}_1, \ldots, \mathbf{x}_k\}$).

Demostración. Si P es un poliedro convexo acotado, entonces $P = \operatorname{conv} \operatorname{ex} P$ por el teorema de Krein–Milman. El conjunto de puntos extremos $\operatorname{ex} P$ está formado por los vértices de P y es un conjunto finito.

Ahora sea $P = \operatorname{conv}\{\mathbf{x}_1, \ldots, \mathbf{x_k}\}$ un politopo convexo. Sin pérdida de generalidad, podemos considerar P como un politopo en $\operatorname{af} P$ y suponer que $\operatorname{int} P \neq \emptyset$. También podemos suponer que $\mathbf{0} \in \operatorname{int} P$ aplicando una traslación. Como hemos observado en la lección sobre conjuntos polares, $\mathbf{0} \in \operatorname{int} P$ implica que el conjunto polar P° es acotado y además se ve que es un poliedro:

$$
\begin{aligned}
P^\circ &= (\operatorname{conv}\{\mathbf{x}_1, \ldots, \mathbf{x}_k\})^\circ \\
&= \{\mathbf{x}_1, \ldots, \mathbf{x}_k\}^\circ \\
&= \left(\bigcup_{1\le i\le k} \{\mathbf{x}_i\}\right)^\circ \\
&= \bigcap_{1\le i\le k} \{\mathbf{x}_i\}^\circ
\end{aligned}
$$

donde $\{\mathbf{x}_i\}^\circ$ son semiespacios cerrados. Entonces, P° es también un poliedro convexo acotado. El mismo razonamiento demuestra que $P^{\circ\circ}$ es un politopo convexo acotado, pero $P^{\circ\circ} = P$. ■

El teorema anterior fue demostrado por Minkowski (Minkowski, H. Allgemeine Lehrsätze über die konvexe Polyeder, Nachr. Ges. Wiss., Göttingen, 1897, 198–219) y otro matemático alemán Hermann Weyl (1885–1955) (Weyl, H. Elementare Theorie der konvexen Polyeder, Comment. Math. Helvetici, 1935, 7). Minkowski y Weyl representaban la escuela de matemáticas de la Universidad de Gotinga, donde también trabajaba Hilbert, que era el director de tesis de Weyl. Al principio de la guerra en 1933 Weyl huyó a los Estados Unidos, donde tuvo plaza en el Instituto de Estudios Avanzados.

Weyl hizo varias contribuciones en geometría (entre otras cosas, dio definiciones rigurosas de las superficies de Riemann y variedades topológicas, estudió representaciones de grupos de Lie compactos, etc.), lógica y física matemática.

No confundir con el matemático francés André Weil (1906–1998). El apellido de Weyl se pronuncia [vail] y el apellido de Weil se pronuncia [veyl].

11. Poliedros en SAGE

Para hacer cálculos con poliedros existen varios programas, y aquí voy a explicar cómo usar SAGE. Pueden descargar este programa y obtener documentación completa en el sitio `http://sagemath.org/`. Para probar los ejemplos de abajo sin instalar el programa (que pesa alrededor de 3 gigabytes), pueden también usar el sitio `https://cloud.sagemath.com/`.

Para definir un politopo $P = \operatorname{conv}\{\mathbf{v}_0, \ldots, \mathbf{v}_k\}$, se puede especificar una lista de puntos. Por ejemplo, si $\mathbf{v}_0 = (1,0,0)$, $\mathbf{v}_1 = (0,1,0)$, $\mathbf{v}_1 = (0,0,1)$, entonces escribimos

```
sage: P = Polyhedron(vertices = [[1,0,0],[0,1,0],[0,0,1]])
sage: P
A 2-dimensional polyhedron in ZZ^3 defined as the convex hull of 3 vertices
```

La lista de vértices de P se obtiene por `P.vertices()`:

```
sage: P.vertices()
(A vertex at (0, 0, 1), A vertex at (0, 1, 0), A vertex at (1, 0, 0))
```

Para obtener las desigualdades y ecuaciones que definen a P, escribimos `P.Hrepresentation()`:

```
sage: P.Hrepresentation()
(An equation (1, 1, 1) x - 1 == 0,
 An inequality (0, -1, -1) x + 1 >= 0,
 An inequality (0, 1, 0) x + 0 >= 0,
 An inequality (0, 0, 1) x + 0 >= 0)
```

Esto quiere decir que $P = \operatorname{conv}\{\mathbf{v}_0, \mathbf{v}_1, \mathbf{v}_2\}$ es la misma cosa que el conjunto en $\mathbb{R}^3$ definido por

$$\begin{aligned} x_1 + x_2 + x_3 - 1 &= 0, \\ -x_2 - x_3 + 1 &\geq 0, \\ x_2 &\geq 0, \\ x_3 &\geq 0. \end{aligned}$$

Para obtener las caras de dimensión d, escribimos `P.faces(d)`:

```
sage: P.faces(0)
(<0>, <1>, <2>)
sage: P.faces(1)
(<0,1>, <0,2>, <1,2>)
sage: P.faces(2)
(<0,1,2>,)
sage: P.faces(3)
()
```

Por ejemplo, `P.faces(2)` nos da `(<0,1,2>,)`, lo cual significa que hay una cara de dimensión 2, a saber $\operatorname{conv}\{\mathbf{v}_0, \mathbf{v}_1, \mathbf{v}_2\}$ (los números $0, 1, 2$ corresponden a los elementos de la lista `P.vertices()`).

Algunos politopos básicos ya están definidos por SAGE. Por ejemplo,

- `polytopes.simplex(d)` es el símplice de dimensión d en $\mathbb{R}^{d+1}$ que tiene como vértices los $d+1$ puntos $\mathbf{v}_i := (0, \ldots, 1, \ldots, 0)$. Por ejemplo, en nuestros ejemplos de arriba, `P = polytopes.simplex(2)`.
- `polytopes.hypercube(d)` es el hipercubo de dimensión d en $\mathbb{R}^d$ que tiene como vértices los 2^d puntos con coordenadas ± 1.
- `polytopes.cube()` es la misma cosa que `polytopes.hypercube(3)`.
- `polytopes.tetrahedron()` es el tetraedro en $\mathbb{R}^3$ de vértices $(0,0,0)$, $(0,1,1)$, $(1,0,1)$, $(1,1,0)$.
- `polytopes.octahedron()` es el octaedro en $\mathbb{R}^3$ de vértices $(\pm 1, 0, 0)$, $(0, \pm 1, 0)$, $(0, 0, \pm 1)$.
- `polytopes.icosahedron()` es el icosaedro en $\mathbb{R}^3$ de vértices:

```
sage: I = polytopes.icosahedron()
sage: I
A 3-dimensional polyhedron in
(Number Field in sqrt5 with defining polynomial x^2 - 5)^3
defined as the convex hull of 12 vertices
sage: I.vertices()
(A vertex at (0, 1/2, 1/4*sqrt5 + 1/4),
 A vertex at (0, -1/2, 1/4*sqrt5 + 1/4),
 A vertex at (1/2, 1/4*sqrt5 + 1/4, 0),
 A vertex at (1/2, -1/4*sqrt5 - 1/4, 0),
 A vertex at (1/4*sqrt5 + 1/4, 0, 1/2),
 A vertex at (1/4*sqrt5 + 1/4, 0, -1/2),
 A vertex at (-1/2, 1/4*sqrt5 + 1/4, 0),
 A vertex at (-1/2, -1/4*sqrt5 - 1/4, 0),
 A vertex at (-1/4*sqrt5 - 1/4, 0, 1/2),
 A vertex at (0, 1/2, -1/4*sqrt5 - 1/4),
 A vertex at (0, -1/2, -1/4*sqrt5 - 1/4),
 A vertex at (-1/4*sqrt5 - 1/4, 0, -1/2))
```

- `polytopes.dodecahedron()` es el dodecaedro en $\mathbb{R}^3$ con 20 vértices.
- Etc. — vean la lista completa en http://doc.sagemath.org/html/en/reference/geometry/sage/geometry/polyhedron/library.html

Para calcular el politopo polar, se usa `P.polar()`. Por ejemplo, podemos comprobar que el politopo polar del cubo es el octaedro y el politopo polar del icosaedro es el dodecaedro:

```
sage: polytopes.cube().polar() == polytopes.octahedron()
True
sage: polytopes.icosahedron().polar() == polytopes.dodecahedron()
True
```

SAGE trata de modo diferente los politopos definidos sobre anillos diferentes. Por ejemplo, el tetraedro en SAGE es definido como la envoltura convexa de los puntos $(0,0,0)$, $(0,1,1)$, $(1,0,1)$, $(1,1,0)$, y entonces es definido sobre $\mathbb{Z}$. SAGE calcula el politopo polar solo cuando $\mathbf{0} \in \operatorname{int} P$ (porque si $\mathbf{0} \notin \operatorname{int} P$, entonces P° no es un conjunto acotado), lo cual no es el caso para el tetraedro:

```
sage: polytopes.tetrahedron().polar()
---------------------------------------------------------------------------
. . . . .
ValueError: The polytope must have the IP property.
```

Pero podemos definir el mismo politopo sobre $\mathbb{Q}$ escribiendo

```
Polyhedron(vertices = [[0,0,1],[0,1,0],[1,0,0],[0,0,0]], base_ring=QQ)
```

En este caso SAGE va a trasladar P de modo que $\mathbf{0} \in \operatorname{int} P$ y después calcular P° (que es también un tetraedro):

```
sage: P = Polyhedron(vertices = [[0,0,1],[0,1,0],[1,0,0],[0,0,0]], base_ring=QQ)
sage: Q = P.polar()
sage: Q.vertices()
(A vertex at (4, 4, 4),
 A vertex at (0, 0, -4),
```

```
 A vertex at (-4, 0, 0),
 A vertex at (0, -4, 0))
```

Para calcular el volumen, hay que escribir `P.volume()`:

```
sage: polytopes.cube().volume()
8
sage: polytopes.tetrahedron().volume()
1/3
sage: polytopes.octahedron().volume()
4/3
sage: polytopes.icosahedron().volume()
5/12*sqrt5 + 5/4
sage: polytopes.dodecahedron().volume()
-176*sqrt5 + 400
```

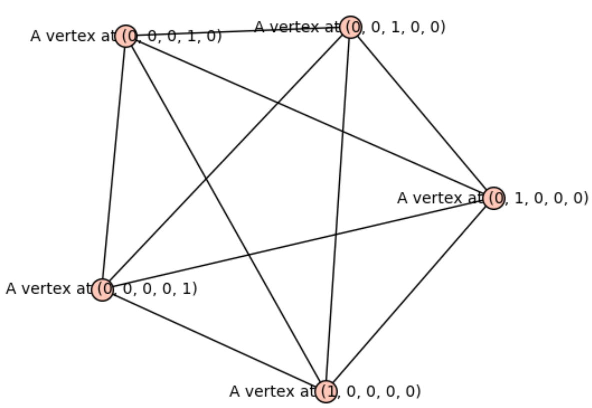
Noten que el volumen se calcula como el volumen en el espacio ambiente. Por ejemplo, el d-símplice tiene volumen 0 porque es considerado como un subconjunto en $\mathbb{R}^{d+1}$. Pero `P.affine_hull()` nos da el mismo politopo pero considerado como un subconjunto en su envolvente afín. Así `polytopes.simplex(`d`).volume() = 0`, pero `polytopes.simplex(`d`).affine_hull().volume() = ` $1/d!$:

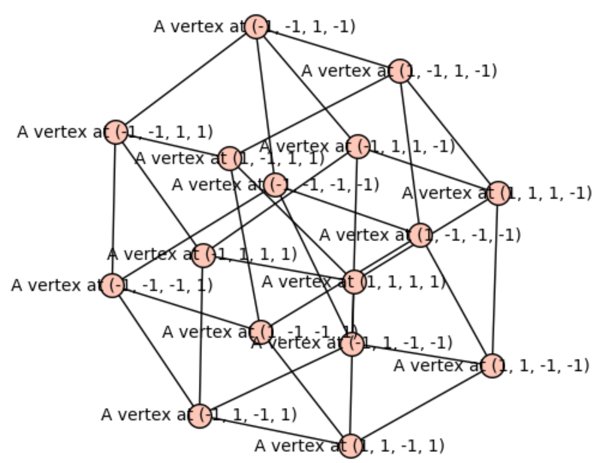
```
sage: polytopes.simplex(2).volume()
0
sage: polytopes.simplex(2).affine_hull().volume()
1/2
sage: polytopes.simplex(3).affine_hull().volume()
1/6
sage: polytopes.simplex(4).affine_hull().volume()
1/24
```

Podemos visualizar el grafo de vértices de un politopo, o el mismo politopo, si tiene dimensión ≤ 4. Por ejemplo, el código

```
sage: polytopes.simplex(4).vertex_graph().graphplot().show()
sage: polytopes.hypercube(4).vertex_graph().graphplot().show()
```

produce las siguientes imágenes:

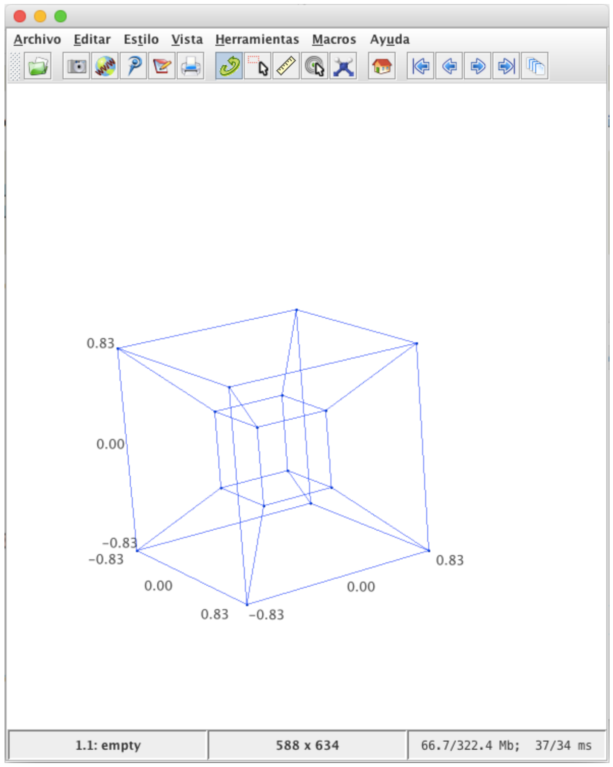




Para ver mejor la estructura del 4-cubo, podemos también escribir

```
sage: polytopes.hypercube(4).show()
```

El resultado es un modelo interactivo:



11.1. Ejercicio. *Descargue SAGE, busque la documentación sobre politopos y pruebe algunos ejemplos.*