Homework 2 - Writing

María del Pilar Cano Vila.

November 21st, 2014.

1 Writing

(1)Let $P \subset \mathbb{R}^d$, $Q \subset \mathbb{R}^e$ be two non-empty polytopes. Prove that the set of faces of the cartesian product polytope $P \times Q = \{(p,q) \in \mathbb{R}^{d+e} : p \in P, q \in Q\}$ exactly equals to the $\{F \times G : F \text{ is a face of } P, G \text{ is a face of } Q\}$. **Conclude that**

$$f_k(P \times Q) = \sum_{\substack{i+j=k \\ i,j \geq k}} f_i(P) f_j(Q) \tag{1}$$

for $k \geq 0$

and use this formula to calculate the entire f-vector of the permutahedron

$$P_n = conv\{(\pi(1), \pi(2), \ldots, \pi(n))^T : \pi \in S_n\} \tag{2}$$

.

Proof. $\subseteq$) Let S be the set of all faces of $P \times Q$, and let $F' \in S$. Then, (Ziegler's definition of a face of a Polytope) $F' = (P \times Q) \cap \{x \in \mathbb{R}^{d+e} : ax = a_0\}$, where $ax \leq a_0$ is valid for $P \times Q$.

Let $x \in F'$, then $x = (p_1, q_1)$ for some $p_1 \in P$ and $q_1 \in Q$, let $a = (a_1, a_2)$ such that $a_1 \in \mathbb{R}^d$ and $a_2 \in \mathbb{R}^e$. So, $ax = a_1p_1 + a_2q_1 = a_0$ where $a_1p_1 = a_{0_1}$ and $a_2q_1 = a_{0_2}$, so $a_{0_1} + a_{0_2} = a_0$.

Claim 1) $a_1p \leq a_{0_1}$ is valid for all $p \in P$ and $a_2q \leq a_{0_2}$ is valid for all $q \in Q$.

By contradiction assume that there exist a $p' \in P$ such that $a_1p' > a_{0_1}$, by definition of $(p', q_1) \in P \times Q$, then $a(p', q_1) \leq a_0$ is valid. On the other hand, $a(p', q_1) = a_1p' + a_2q_1 = a_1p' + a_{0_2} > a_{0_1} + a_{0_2} = a_0$, which contradicts that $a(p', q_1) \leq a_0$ is valid. Therefore $a_1p \leq a_0$ is valid in P.

To show that $a_2q \leq a_{0_2}$ is valid for all $q \in Q$ is analogous.

Claim 2) For all $x = (p,q) \in F'$, $a_1p = a_{0_1}$ and $a_2q = a_{0_2}$ holds.

Notice that if the inequality does not hold for p', then does not hold for q' too, and viceversa. By contradiction assume that there exist a $x' = (p', q') \in F'$ such that $a_1p' \neq a_{0_1}$. Then $a_1p' < a_{0_1}$ or $a_1p' > a_{0_1}$, by Claim 1 we know that $a_1p' > a_{0_1}$ cannot happen, so $a_1p' < a_{0_1}$. We know by definition of F' that $a_1p' + a_2q' = a_0$, so $a_2q' > a_{0_2}$ which contradicts Claim 1. Hence $a_1p = a_{0_1}$ and $a_2q = a_{0_2}$ for all $x = (p,q) \in F'$.

If we assume that $a_2q' \neq a_{0_2}$, the proof is the same.

By Claim 1, there exist a face F in P such that $F = P \cap \{p \in \mathbb{R}^d : a_1p = a_{0_1}\}$ and a face G in Q such that $G = Q \cap \{q \in \mathbb{R}^e : a_2q = a_{0_2}\}$. By Claim 2, for all $x \in F'$, $x \in F \times G$, then $F' \subseteq F \ \times Q$.

Now, let $x = (p,q) \in F \times Q$, then $p \in P$ and $q \in Q$ such that $a_1p = a_{0_1}$ and $a_2q = a_{0_2}$, so, $ax = a_1p + a_2q = a_{0_1} + a_{0_2} = a_0$, thus $x \in F'$. Therefore $F \times G \subseteq F'$.

Hence, $F' = F \times G$, and so $F' \in \{F \times G : F \text{ is a face of } P, G \text{ is a face of } Q\}$.
Therefore, $S \subseteq \{F \times G : F \text{ is a face of } P, G \text{ is a face of } Q\}$.

$\supseteq$) Let $F \times G \in \{F \times G : F \text{ is a face of } P, G \text{ is a face of } Q\}$, then $F = P \cap \{p \in \mathbb{R}^d : a_1 p = a_{0_1}\}$ and $G = Q \cap \{q \in \mathbb{R}^e : a_2 q = a_{0_2}\}$, where $a_1 p \leq a_{0_1}$ is valid in P and $a_2 q \leq a_{0_2}$ is valid in Q. Let $a = (a_1, a_2) \in \mathbb{R}^{d+e}$ and $x = (p, q) \in P \times Q$, since the previous inequalities holds for P and Q respectively, then $ax = a_1 p + a_2 q \leq a_{0_1} + a_{0_2} = a_0$ is valid for $P \times Q$, for some $a_0 \in \mathbb{R}$. Then $F' = (P \times Q) \cap \{x \in \mathbb{R}^{d+e} : ax = a_0\}$ is a face of $P \times Q$. By definition of $F \times G$, clearly $F \times G \subseteq F'$.
Now, let $x = (p, q) \in F'$, then $ax = a_0$, so $a_1 p + a_2 q = a_0$.

Claim 3) $a_1 p = a_{0_1}$ and $a_2 q = a_{0_2}$.
By contradiction assume that this is not true, then $a_1 p > a_{0_1}$ or $a_2 q > a_{0_2}$, since $p \in P$ and $q \in Q$, then, both cases cannot happen since $a_1 p \leq a_{0_1}$ for all $p \in P$ and $a_2 q \leq a_{0_2}$ for all $q \in Q$. Thus, $a_1 p = a_{0_1}$ and $a_2 q = a_{0_2}$.
By Claim 3, $x \in F \times G$, so $F' \subseteq F \times G$. Therefore, $F' = F \times G$.
So, $\{F \times G : F \text{ is a face of } P, G \text{ is a face of } Q\} \subseteq S$.

Hence $\{F \times G : F \text{ is a face of } P, G \text{ is a face of } Q\} = S$.

□

Now that we proved that, all the faces of $P \times Q$ are of the form $F \times G$ where F is a face of P and G a face of Q, we can calculate the f-vector of $P \times Q$. So,
$f_k(P \times Q) = (\#\text{faces of dimension } k) = (\# F \times G \text{ such that } dim(F) + dim(G) = k) =$
$(\sum_{i+j=k,i,j\geq 0}(\#F \times \#Q)$ such that $dim(F) = i$ and $dim(G) = j) = \sum_{i+j=k,i,j\geq 0} f_i(P) f_j(Q)$.

In order to calculate the f-vector of the permutahedron first, notice that P_n is a polytope of dimension $n-1$, also, we know that it is the conv$\{(\pi(1), \pi(2), \ldots, \pi(n))^T : \pi \in S_n\}$, so, $f_0(P_n) = n!$ two vertices are adjacent if there is only one change, i.e., a edge of P_n is the n_2 the cross product of P_1 times P_2, since we are only permutating two entries of our vector, then $f_1(P_n)$ are all the ways to choose the partition of $n-1$ non-empty subsets of $[n]$, which is equal to $f_1(P_n) = (\text{all the ways to choose the partition})(f_1(P_2 \times P_1 \times \ldots \times P_1)) = \frac{n!(n-1)}{2} f_1(P_2) = \frac{n!(n-1)}{2}$. We also know that the facets of P_n are of the form $P_m \times P_{n-m}$, for all $m < n$, so, we are choosing a partition of $[n]$ into to sets, which if we count all the forms to choose this set are all the non-empty subsets of $[n]$ without $\emptyset$ and $[n]$, so $f_{n-2}(P_n) = 2^n - 2$.
So, following the same reasoning we get that a k-face of P_n is given by a $n-k$ partition of $[n]$, so a face of degree k in P_n is of the form $P_{t_1} \times P_{t_2} \times \ldots P_{t_{n-k}}$ and t_1 can goes from 1 to $k+1$, since we have to have the other sets with at least one element and $k+1+n-k-1 = n$. So, $P_{t_2} \times P_{t_3} \times \ldots \times P_{t_{n-k}}$ is a $k - t_1 + 1$ face of P_{n-t_1}. Then
$f_k(P_n) = \sum_{t_1=1}^{k+1}(\text{all the ways to choose the set with } t_1 \text{ element in [n]}) f_{k-t_1+1}(P_{t_2} \times P_{t_3} \times \ldots \times P_{t_{n-k}}) =$
$\sum_{t_1=1}^{k+1} \left(\begin{array}{c} n \\ k \end{array} \right) f_{k-t_1+1}(P_{n-t_1})$.

2) A *Lattice polytope* is the convex hull of finitely many vertices many vertices with integer coordinates. Two lattice polytopes $P, Q \subset \mathbb{R}^d$ are *lattice equivalent* or *latticee isomorphic* if there exist $A \in \mathbb{Z}^{d \times d}$ with $|det(A)| = 1$ and $b \in \mathbb{Z}^d$ such that $Q = f(P)$ for the affine mat $f(x) = Ax + b$. (How is this different from demanding that P and Q be $\mathrm{Sl}_d(\mathbb{Z})$-equivalent?)

If we demand that P and Q be $\mathrm{Sl}_d(\mathbb{Z})$-equivalent, then A should have determinant 1, if P and Q are lattice equivalent and the determinant of A is negative, since the absolute value of A is 1, then P and Q have the same area, but since the determinant of A is -1, then the area of Q is $-$ times the area of P, this means, that the points of Q change the orientation, this means that it is some reflection and some move of P. So, if we want that P and Q be $\mathrm{Sl}_d(\mathbb{Z})$-equivalent, then P is Q move in someway, without reflections.

(a) Prove that the *modular group* $\mathbf{Sl}_2(\mathbb{Z})$ is generated by the elements $S = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$ and $T = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$, i.e., any $A \in \mathbf{Sl}_2(\mathbb{Z})$ is expressible as a product of matrices S and T.

Proof. Notice that $det(S) = 1$ and $det(T) = 1$, so, $S, T \in \mathrm{Sl}_2(\mathbb{Z})$, since $\mathrm{Sl}_2(\mathbb{Z})$ is a group, then if $A \in \mathrm{Sl}_2(\mathbb{Z})$, then $AS, AT \in \mathrm{Sl}_2(\mathbb{Z})$ and $A^{-1} \in \mathrm{Sl}_2(\mathbb{Z})$.

Also, notice that if $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ then $AS = \begin{pmatrix} b & -a \\ d & -c \end{pmatrix}$, $T^n = \begin{pmatrix} 1 & n \\ 0 & 1 \end{pmatrix}$, $T^{-n} = \begin{pmatrix} 1 & -n \\ 0 & 1 \end{pmatrix}$ and $AT^n = \begin{pmatrix} a & b+an \\ c & d+cn \end{pmatrix}$.

So that,
$ST^n = \begin{pmatrix} 0 & -1 \\ 1 & n \end{pmatrix}$, $T^nS = \begin{pmatrix} -n & 1 \\ -1 & 0 \end{pmatrix}$ and $ST^nS = \begin{pmatrix} 1 & 0 \\ -n & 1 \end{pmatrix}$ and $S^2 = -I_2$.

Let $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ be in $\mathrm{Sl}_2(\mathbb{Z})$, let us proof that A is a product of T's and S's. Since $A \in \mathrm{Sl}_2(\mathbb{Z})$, then $det(A) = 1$, then $gdc(a, b) = 1$ and $gdc(c, d) = 1$.

If $|a| < |b|$, then $b = an + r$ for some n, where $0 < |r| < |a|$, so $AT^{-n} = \begin{pmatrix} a & r \\ c & d - cn \end{pmatrix}$, if $|a| > |b|$ we can switch them applying AS and do the same. We can do these steps until we get to something of the form $AT^{-n_1}ST^{-n_2}\dots ST^{-n_k} = \begin{pmatrix} q & 1 \\ p & t \end{pmatrix}$, we can get to this form since $gdc(a, b) = 1$ and by the division theorem (it is implicit that we are using euclides's algorithm), then we switch the columns multiplying by S and remove q in the entries and we obtain $AT^{-n_1}ST^{-n_2}\dots ST^{-n_k}ST^q = \begin{pmatrix} -1 & 0 \\ m & -1 \end{pmatrix}$, this the other entry $(2, 2)$ going to be also -1, since the determinant is 1. But this matrix is of the form $ST^{-m}S$. So, we get that $AT^{-n_1}ST^{-n_2}\dots ST^{-n_k}ST^q = ST^{-m}S$, hence $A = ST^{-m}ST^{-q}S^{-1}T^{n_k}S^{-1}\dots Tn_2S^{-1}T^{n_1}$.

Therfore, S and T generates $\mathrm{Sl}_2(\mathbb{Z})$.

□

(b) Interpret the preceding result geometrically, and use it to classify the lattice polygons with (i) no, (ii) exactly one strictly interior lattice point up to lattice equivalence.

S is a rotation of $\pi/2$ in the origin, since $S = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$ and $cos(\pi/2) = 0$ and $sin(\pi/2) = 1$, you can also notice this since $S^4 = I$, and T^ϵ the only thing that it is doing is translating to the left or right depending of the sign of ϵ.

In order to classify the lattice polygons with (i) no, (ii) exactly one strictly interior lattice point up to lattice equivalence, first notice that if a lattice polytopes has a strictly lattice point, then this only can by a triangle, square or a pentagon, otherwise the polygon is not convex. If the lattice polytope does not have interior lattice points, then this has to be in the "corridors" of the lattice,i.e, in between to lattice lines, as in the figure 2, and in order to be convex, can only be a triangle or a quadrilateral.

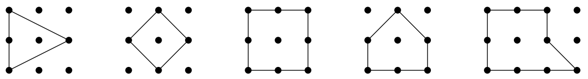

Figure 1: The last polygon is not a polytope and has strictly one interior lattice point.

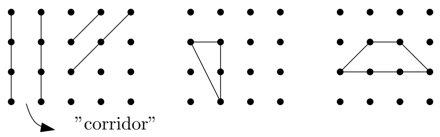

Figure 2: With no interior lattice point.

2 Software

(3) Explain the difference between a public/private key pair of ssh and a public/private keay par for gpg. Gather information about the recommended keysizes, and explain briefly the advantages and disadvantages of the gpg software, making special mention of the latest versions.

On one hand, A pair of public/private pair of *ssh* key is used in order to connect to some server or several serves, this works for verified if is you who is trying to enter to this server, this server is like a share server, as what we do in github, if we want to enter to the desecrete-geometry, git first needs to verified that it is you who is trying to access, so it checks if you ssh-key is the same, and you only put it once.

On the other hand, a public/private *gpg* pair key is used in order to encrypt files, and is encrypted by the public key, so if someone wants to send you a encrypted message, can do it by using your public key, and send it to you, and the only way to decrypted is using your private key.

So, ssh-key is used to authenticate when a user wants to enter to some server, and the gpg key is used to encrypt and decrypt messages.

The minimum size that requires the gpg key is 1024 bits and the maximum is 4096 bits, but it is not recommended to use 1024 bits since this is a weak key and can be corrupt easier, and 4096 it is also not recommended since the decrypt and encrypt take more time, so the recommended size is something in between which is 2048 bits.

An advantage of using the gig software is that is a free software but from this, GnuPG do not support some patent encrypt licensed since, GnuPG uses non patent licensed algorithm as ElGamal, CASTS. For instance GnuPG did not support IDEA, but since 2012 that the las patent of IDEA expired, and the version 2.1 support elliptic curve cryptography which it makes it faster and we can use greater keys. Another advantage is that if you have lost someway your secrete key, you do not have to generate a new one, you can revoked it and get a new one without generating your key, since you only have a sub key. A disadvantage could be that using expiration dates with the subkeys means that the subkeys must be extended prior to expiration, or new subkeys issued if they are allowed to expire. Another disadvantage is that other people who use your public key may receive errors about your key being expired if they do not regularly update their GPG keyrings based on the public keyservers.