2. Problemas de mínimos cuadrados: Resolución secuencial y paralela

Bibliografía específica

G.H.Golub and C.F. Van Loan.
"Matrix Computations"
The Johns Hopkins Univ. Press, 3rd Ed. Baltimore, 1996.

E.Anderson, Z.Bai, J.Dongarra
Generalized-QR-Factorization-and-Its-Applications
Linear Algebra and its Applications. Vol.162-164. Pp.243-271 (1992)

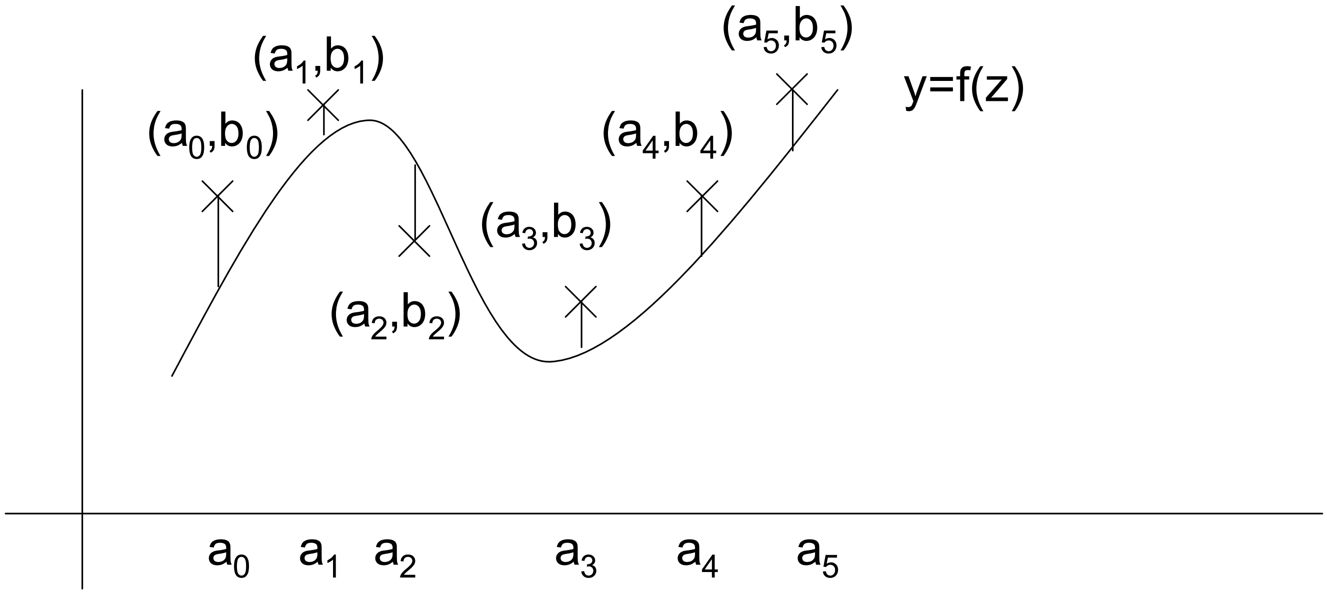
El Problema Lineal de Mínimos Cuadrados (PLMC)

Dada la matriz $A \in \Re^{mxn}, m \geq n,$ de rango completo,

y un vector $b \in \Re^{m}$ encontrar un $x_0 \in \Re^{n}$ tal que

$$\left\| Ax_0 - b \right\|_2 = \min_{x \in \Re^n} \left\| Ax - b \right\|_2$$

Aplicaciones: Ajuste de curvas



$$f(z) = x_0 + x_1 z + x_2 z^2 + x_3 z^3$$

Calcular los coeficientes del polinomio en z para que $\sum_{i=0}^{m-1} (f(a_i) - b_i)^2$ con m=5, sea mínimo

con $$f(a_i) - b_i = x_0 + x_1 a_i + x_2 a_i^{\ 2} + x_3 a_i^{\ 3} - b_i$$

Definamos las siguientes matrices y vectores:

$$A = \begin{bmatrix} 1 & a_0 & a_0^{\,2} & a_0^{\,3} \\ 1 & a_1 & a_1^{\,2} & a_1^{\,3} \\ 1 & a_2 & a_2^{\,2} & a_2^{\,3} \\ 1 & a_3 & a_3^{\,2} & a_3^{\,3} \\ 1 & a_4 & a_4^{\,2} & a_4^{\,3} \end{bmatrix}; \; x = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix}; \; b = \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix};$$

Obsérvese que:

$$f(a_i) - b_i = x_0 + x_1 a_i + x_2 a_i^{\,2} + x_3 a_i^{\,3} - b_i = (Ax - b)_i = A_{(i)} x - b_i$$

Por tanto:

$$\sum_{i=0}^{m-1} (f(a_i) - b_i)^2 = \left\| Ax - b \right\|_2^{\,2}$$

$$\min_x \sum_{i=0}^{m-1} (f(a_i) - b_i)^2 = \min_x \left\| Ax - b \right\|_2^{\,2}$$

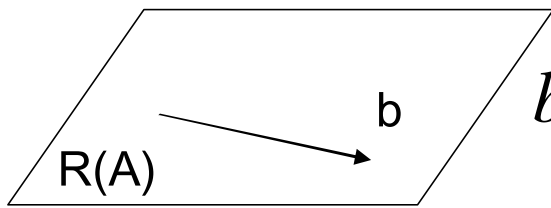
Resolución del PLMC mediante las Ecuaciones normales

$$\min_{x\in\Re^n} \left\| Ax - b \right\|_2 \geq 0$$

Si imponemos el valor más pequeño de los posibles

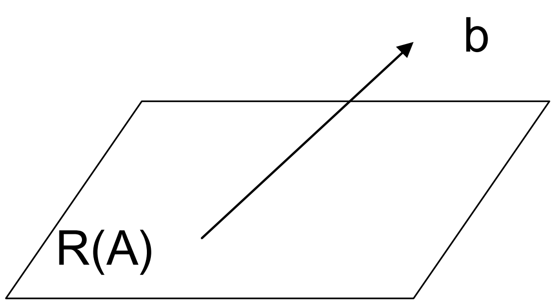
$$\left\| Ax - b \right\|_2 = 0 \Rightarrow Ax = b \Rightarrow b \text{ es una C.L. de las columnas de } A$$

Posibilidades:



$$b \in R(A) \qquad \left\| Ax_0 - b \right\|_2 = 0$$

Solución exacta

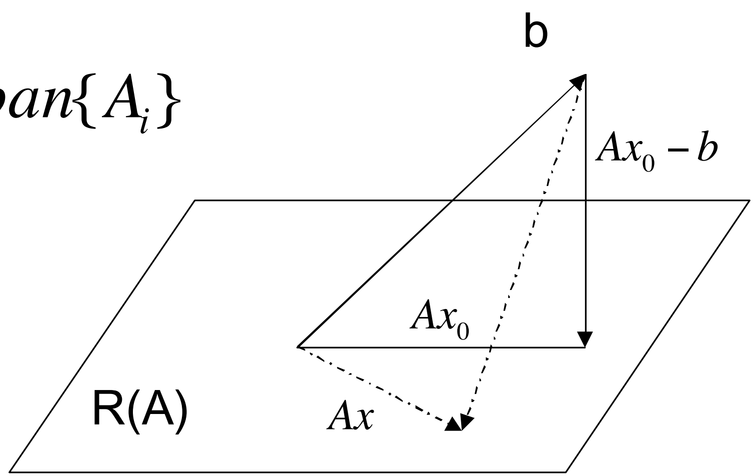


$$b \notin R(A) \qquad \left\| Ax_0 - b \right\|_2 > 0$$

Solución de mínimos cuadrados

Resolución del PLMC mediante las Ecuaciones Normales

$$R(A) = span\{A_i\}$$



Solución de mínimos cuadrados: $$(Ax_0 - b) \perp R(A)$$

$$\text{Para } i = 0,1,\ldots,n-1$$

$$A_i^T (Ax_0 - b) = 0$$

En forma matricial:

$$A^T (Ax_0 - b) = 0 \Leftrightarrow A^T A x_0 = A^T b$$

Ecuaciones normales del PLMC:

Resolución del PLMC mediante las Ecuaciones Normales

Problema con las Ecuaciones normales del PLMC:

$$A^T(Ax_0 - b) = 0 \Leftrightarrow A^T A x_0 = A^T b$$

$$\text{Sea } A = \begin{bmatrix} 1 & 1 \\ \mu & 0 \\ 0 & \mu \end{bmatrix}, \text{tal que } fl(1+\mu^2) = 1$$

$$\text{Entonces } fl(A^T A) = fl(\begin{bmatrix} 1 & \mu & 0 \\ 1 & 0 & \mu \end{bmatrix}\begin{bmatrix} 1 & 1 \\ \mu & 0 \\ 0 & \mu \end{bmatrix}) = \begin{bmatrix} fl(1+\mu^2) & 1 \\ 1 & fl(1+\mu^2) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

Obsérvese que *rank(A)*=2 mientras que *rank(fl(*A^T*A))=*1.
fl(A^T*A)* no es invertible y el problema no se puede resolver aunque *A* es de rango completo-
Al hacer la operación *fl(*A^T*A)* se ha perdido información

Transformaciones que hacen 0's

$$Mx = \begin{bmatrix} m & m & m & m \\ m & m & m & m \\ m & m & m & m \\ m & m & m & m \end{bmatrix} \begin{bmatrix} x \\ x \\ x \\ x \end{bmatrix} = \begin{bmatrix} * \\ 0 \\ 0 \\ 0 \end{bmatrix}; \quad Mx = \begin{bmatrix} m & m & m & m \\ m & m & m & m \\ m & m & m & m \\ m & m & m & m \end{bmatrix} \begin{bmatrix} x \\ x \\ x \\ x \end{bmatrix} = \begin{bmatrix} * \\ 0 \\ x \\ x \end{bmatrix};$$

Ejemplos: Matrices de Gauss, Matrices de Householder, Matrices de Givens

Transformaciones ortogonales

$$Q \in \Re^{nxn} (\mathrm{Q} \in C^{nxn}) \ \text{ tal que } Q^T Q = I (Q^H Q = I) \Leftrightarrow Q^T = Q^{-1} (Q^H = Q^{-1})$$

Ejemplos: Matrices de Householder, Matrices de Givens

Propiedad de las transformaciones ortogonales:

- **Conservan la 2-norma vectorial**

$$\|Qx\|_2 = \sqrt{(Qx)^T (Qx)} = \sqrt{x^T Q^T Q x} = \sqrt{x^T x} = \|x\|_2$$

Resolución del PLMC mediante la Descomposición QR

Sea la matriz $A \in \Re^{mxn}, m \geq n,$ de rango completo, $Q \in \Re^{mxm}$, ortogonal,

$R \in \Re^{mxn}$, triangular superior, con $R = \begin{bmatrix} R_1 \\ 0 \end{bmatrix}$, y supongamos que

$A = QR$ (Descomposición QR de A)

$$\|Ax-b\|_2^2 = \|Q^T(Ax-b)\|_2^2 = \|Q^T Ax - Q^T b\|_2^2 = \|Rx - Q^T b\|_2^2 =$$

$$= \left\| \begin{bmatrix} R_1 \\ 0 \end{bmatrix} x - Q^T b \right\|_2^2 = \|R_1 x - c_1\|_2^2 + \|-c_2\|_2^2, \text{con } c = Q^T b = \begin{array}{c} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} \end{array} \begin{array}{l} n \\ m-n \end{array}$$

$$\|Ax-b\|_2^2 = \|R_1 x - c_1\|_2^2 + \|-c_2\|_2^2, \text{ y } \|-c_2\|_2^2 \text{ no depende de } x$$

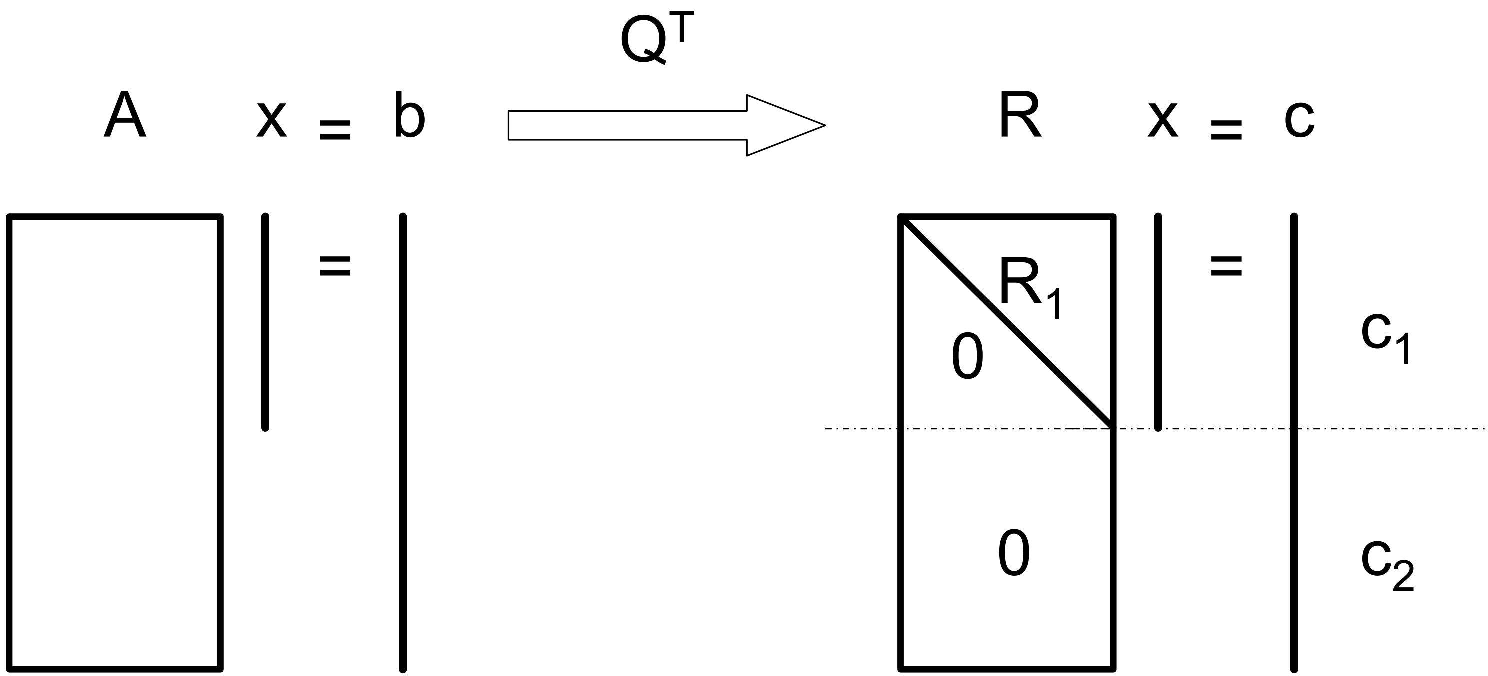
Por tanto, el valor mínimo de

$$\|Ax-b\|_2^2 \text{ se alcanza cuando } \|R_1 x_0 - c_1\|_2^2 = 0 \Leftrightarrow x_0 = R_1^{-1} c_1$$

$$\text{y} \quad \min_{x \in \Re^n} \|Ax-b\|_2^2 = \|-c_2\|_2^2$$

Resolución del PLMC mediante la Descomposición QR

Si A=QR (Descomposición QR)

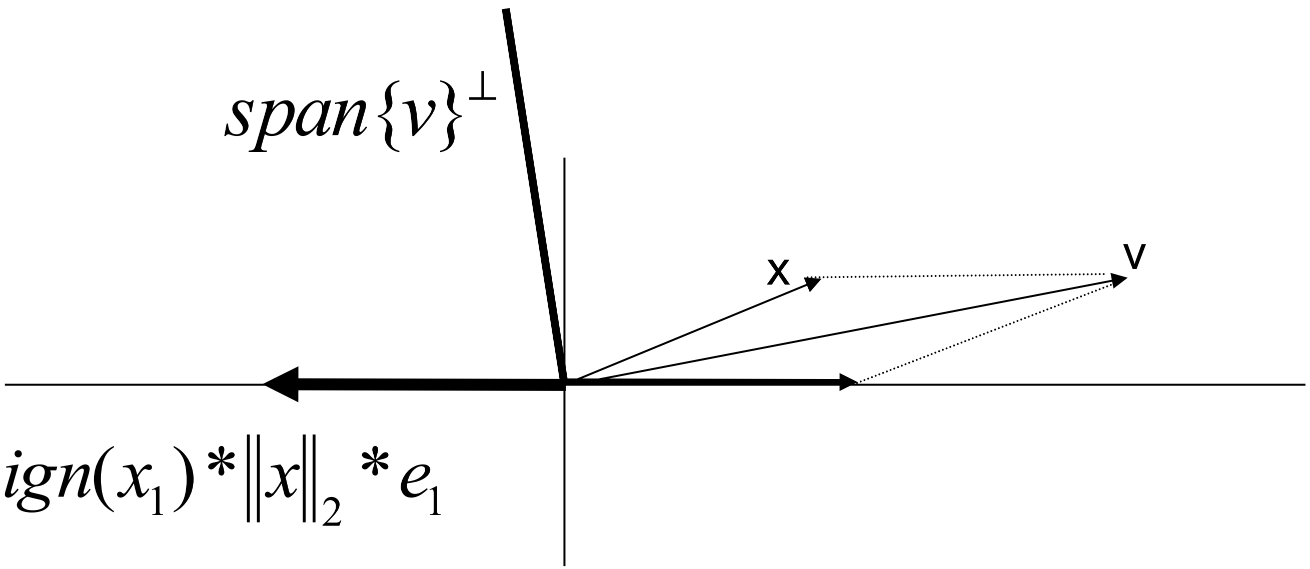


Matrices (reflexiones) de Householder

$$P = I - \frac{2vv^T}{v^T v}, v \neq 0$$

Propiedades:

- Simétricas
- Ortogonales
- Involutivas
- Permiten hacer ceros en las componentes de un vector



$$Px = -sign(x_1) * \|x\|_2 * e_1$$

Permiten hacer ceros en las componentes de un vector

$Si\ x \in \Re^n, x \neq e_1$ y $v = x + sign(x_1) * \|x\|_2 * e_1$

entonces

$$Px = x - \frac{2vv^T x}{v^T v} = x - \frac{(v^T x)}{(\frac{v^T v}{2})} v = x - v = x - (x + sign(x_1) * \|x\|_2 * e_1) = -sign(x_1) * \|x\|_2 * e_1$$

ya que

$$(v^T x) = (x + sign(x_1) * \|x\|_2 * e_1)^T x = \|x\|_2 (\|x\|_2 + sign(x_1) * x_1)$$

$$(\frac{v^T v}{2}) = \frac{(x + sign(x_1) * \|x\|_2 * e_1)^T (x + sign(x_1) * \|x\|_2 * e_1)}{2} =$$

$$= \frac{\|x\|_2^{\ 2} + 2sign(x_1) * \|x\|_2 * x_1 + \|x\|_2^{\ 2}}{2} = \|x\|_2 (\|x\|_2 + sign(x_1) * x_1)$$

Ejemplo

$$x = \begin{bmatrix} 2 \\ 4 \\ 1 \end{bmatrix}; \Rightarrow v = x + sign(x_1) * \|x\|_2 * e_1; \Rightarrow v = \begin{bmatrix} 6.5826 \\ 4 \\ 1 \end{bmatrix};$$

$$P = I - \frac{2vv^T}{v^T v}, v \neq 0 \qquad P = \begin{bmatrix} -0.4364 & -0.8729 & -0.2182 \\ -0.8729 & 0.4696 & -0.1326 \\ -0.2182 & -0.1326 & 0.9668 \end{bmatrix}$$

$$y = Px = \begin{bmatrix} -4.5826 \\ 0 \\ 0 \end{bmatrix}; \qquad norm(y) = norm(x) = 4.5826$$

Algoritmo de triangularización de Householder

Para $j = 0,1,\ldots,n\text{-}1$

Calcula la Matriz de Householder, H_j, que hace ceros en la columna j

Para $k = j, j+1,\ldots,n\text{-}1$

Modifica la columna *k:* $A_k \leftarrow H_j * A_k$

$$\begin{array}{c}
\begin{array}{cccccc} & & j & & & \end{array} \\
\begin{bmatrix}
x & x & x & x & x & x \\
 & x & x & x & x & x \\
 & & x & x & x & x \\
 & & x & \mathrm{x} & \mathrm{x} & \mathrm{x} \\
 & & x & \mathrm{x} & \mathrm{x} & \mathrm{x} \\
 & & x & \mathrm{x} & \mathrm{x} & \mathrm{x}
\end{bmatrix} \\
\begin{array}{cccccc} & & k & k & k & k \end{array}
\end{array}$$

$$\text{Coste}: \quad 2n^2\left(m - \frac{n}{3}\right) \ \textit{Flops}$$

Construcción del algoritmo

$v = x + sg(x_1)\,||x||_2 e_1$
$\beta = (v^t v)/2 = sg(x_1)\,||x||_2 (sg(x_1)\,||x||_2 + x_1)$

$$\begin{pmatrix} x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \\ x & x & x & x & x \end{pmatrix} \Rightarrow \begin{pmatrix} x & x & x & x & x \\ x & x & x & x & x \\ x & 0 & x & x & x \\ x & 0 & x & x & x \\ x & 0 & x & x & x \end{pmatrix}$$

<u>Procedimiento Norm(j)</u>

$$\rho_j = \operatorname{sgn}(a_{jj})\left(\sum_{i=j}^{m-1} a_{ij}^{\,2}\right)^{1/2}$$

$$v_j = [0,0,\ldots,0, a_{jj} + \rho_j, a_{j+1,j}, \ldots, a_{m-1,j}]^T$$

$$\beta_j = \rho_j(\rho_j + a_{jj})$$

Coste: 2(m-j) Flops

Resultado de aplicar una Matriz de Householder, P, a otro vector y distinto del x que ha servido para calcularla

$$Py = y - \gamma v \qquad \gamma = (1/\beta)(v^T y) \text{ y } \beta = (v^T v)/2$$

Procedimiento Fact(j,k)

$$\gamma_{jk} = (1/\beta_j)\sum_{i=j}^{m-1}(v_j)_i a_{ik}$$

Coste: 2(m-j) Flops

Procedimiento Cmod(j,k)

$$Para\ i = j, j+1, \ldots, m-1$$
$$a_{ik} = a_{ik} - \gamma_{jk}(v_j)_i$$

Coste: 2(m-j) Flops

$$\begin{pmatrix} x & x & x & x & x \\ 0 & \boxed{x} & x & x & x \\ 0 & \boxed{x} & x & x & x \\ 0 & \boxed{x} & x & x & x \\ 0 & \boxed{x} & x & x & x \end{pmatrix} \Rightarrow \begin{pmatrix} x & x & x & x & x \\ 0 & x & \boxed{x} & \boxed{x} & \boxed{x} \\ 0 & 0 & \boxed{x} & \boxed{x} & \boxed{x} \\ 0 & 0 & \boxed{x} & \boxed{x} & \boxed{x} \\ 0 & 0 & \boxed{x} & \boxed{x} & \boxed{x} \end{pmatrix}$$

Algoritmo de Triangularización Ortogonal (Householder)

Para $j = 0,1,\ldots,n\text{-}1$

  Calcula la Matriz de Householder, H_j, que hace ceros en la columna j

  Para $k = j, j+1,\ldots,n\text{-}1$

    Modifica la columna k: $A_k \leftarrow H_j * A_k$

Para j=0,1,…,n-1
  norm(j)
  Para k=j,j+1,…,n-1
    fact(j,k)
    cmod(j,k)

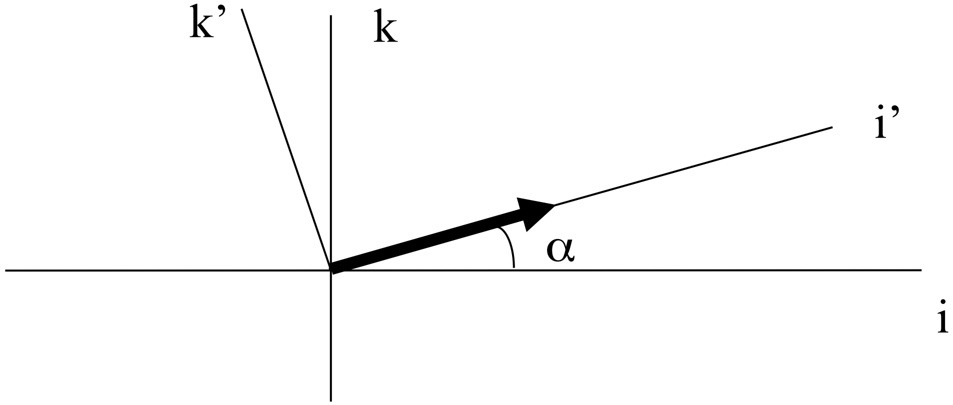
$$\text{Coste}: \ 2n^2\left(m - \frac{n}{3}\right) \ Flops$$

Rotaciones de Givens

$$G_{ik} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & c & 0 & s & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & -s & 0 & c & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}; \qquad G_{ik} = \begin{cases} G(i,i) = G(k,k) = c = \cos\alpha \\ G(i,k) = -G(k,i) = s = \sin\alpha \\ G(r,t) = 1,\ \mathrm{i} \neq r = s \neq k \\ G(r,t) = 0,\ r\text{esto de casos} \end{cases}$$

Propiedades:

- Ortogonales
- Permiten hacer ceros en una componente de un vector



Rotaciones de Givens

Caso 2x2

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}; \quad G = \begin{bmatrix} c & s \\ -s & c \end{bmatrix}; \quad c = \frac{x_1}{\sqrt{x_1^{\,2} + x_2^{\,2}}}; s = \frac{x_2}{\sqrt{x_1^{\,2} + x_2^{\,2}}}$$

$$Gx = \begin{bmatrix} c & s \\ -s & c \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} cx_1 + sx_2 \\ -sx_1 + cx_2 \end{bmatrix} = \begin{bmatrix} \sqrt{x_1^{\,2} + x_2^{\,2}} \\ 0 \end{bmatrix}; \quad c = \frac{x_1}{\sqrt{x_1^{\,2} + x_2^{\,2}}}; s = \frac{x_2}{\sqrt{x_1^{\,2} + x_2^{\,2}}}$$

Caso nxn

$$x = \begin{bmatrix} x_1 \\ . \\ x_i \\ . \\ x_k \\ . \\ . \\ x_n \end{bmatrix}; \quad G = \begin{bmatrix} 1 & 0. & . & . & & . & 0 & 0 \\ . & . & & & & & . & . \\ 0 & & c & & s & & . & 0 \\ 0 & & & 1 & & & & \\ 0 & & -s & & c & & & \\ . & & & & & . & . & . \\ . & & & & & & 1 & 0 \\ 0 & 0 & . & . & . & . & 0 & 1 \end{bmatrix}; \quad c = \frac{x_i}{\sqrt{x_i^{\,2} + x_k^{\,2}}}; s = \frac{x_k}{\sqrt{x_i^{\,2} + x_k^{\,2}}}$$

$$Gx = \begin{bmatrix} 1 & 0 & . & . & & . & 0 & 0 \\ . & . & & & & & . & . \\ 0 & & c & & s & & . & 0 \\ 0 & & & 1 & & & & \\ 0 & & -s & & c & & & \\ . & & & & & . & . & . \\ . & & & & & & 1 & 0 \\ 0 & 0 & . & . & . & . & 0 & 1 \end{bmatrix}\begin{bmatrix} x_1 \\ . \\ x_i \\ . \\ x_k \\ . \\ . \\ x_n \end{bmatrix} = \begin{bmatrix} x_1 \\ . \\ \sqrt{x_i^{\,2} + x_k^{\,2}} \\ . \\ 0 \\ . \\ . \\ x_n \end{bmatrix}$$

Ejemplos

$$x = \begin{bmatrix} 2 \\ 3 \\ 6 \\ 7 \\ 9 \end{bmatrix};$$

$$\mathrm{t} = \frac{2}{7} = 0.2857;\ \ s = 1/\sqrt{1+t^2} = 0.9615;$$

$$c = s * t = 0.2747$$

$$G = \begin{bmatrix} 0.2747 & 0 & 0 & 0.9615 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ -0.9647 & 0 & 0 & 0.2747 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$y = Gx = \begin{bmatrix} 0.2747 & 0 & 0 & 0.9615 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ -0.9647 & 0 & 0 & 0.2747 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \\ 6 \\ 7 \\ 9 \end{bmatrix} = \begin{bmatrix} 7.2801 \\ 3 \\ 6 \\ 0 \\ 9 \end{bmatrix}$$

$$x = \begin{bmatrix} 2 \\ 3 \\ 6 \\ 7 \\ 9 \end{bmatrix};$$

$$\mathrm{t} = \frac{3}{6} = 2;\ \ s = 1/\sqrt{1+t^2} = 0.8944;$$

$$c = s * t = 0.4472$$

$$G_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0.4472 & 0.8944 & 0 & 0 \\ 0 & -0.8944 & 0.4472 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$y = G_2 x = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0.4472 & 0.8944 & 0 & 0 \\ 0 & -0.8944 & 0.4472 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \\ 6 \\ 7 \\ 9 \end{bmatrix} = \begin{bmatrix} 2 \\ 6.7082 \\ 0 \\ 7 \\ 9 \end{bmatrix}$$

Descomposición QR vía Rotaciones de Givens

$$A=\begin{bmatrix} a & a & a & a\\ a & a & a & a\\ a & a & a & a\\ a & a & a & a\end{bmatrix}\Rightarrow\begin{bmatrix} a & a & a & a\\ a & a & a & a\\ b & b & b & b\\ 0 & b & b & b\end{bmatrix}\Rightarrow\begin{bmatrix} a & a & a & a\\ b & b & b & b\\ 0 & b & b & b\\ 0 & b & b & b\end{bmatrix}\Rightarrow\begin{bmatrix} b & b & b & b\\ 0 & b & b & b\\ 0 & b & b & b\\ 0 & b & b & b\end{bmatrix}$$

Coste: 2n³ Flops

$$G_{1,1}\ldots * G_{n-2,1} * G_{n-1,1} * A = R_1$$

$$R_1=\begin{bmatrix} b & b & b & b\\ 0 & b & b & b\\ 0 & b & b & b\\ 0 & b & b & b\end{bmatrix}\Rightarrow\begin{bmatrix} b & b & b & b\\ 0 & b & b & b\\ 0 & c & c & c\\ 0 & 0 & c & c\end{bmatrix}\Rightarrow\begin{bmatrix} b & b & b & b\\ 0 & c & c & c\\ 0 & 0 & c & c\\ 0 & 0 & c & c\end{bmatrix}$$

$$G_{2,2}\ldots * G_{n-2,2} * G_{n-1,2} * R_1 = R_2$$

$$R_2=\begin{bmatrix} b & b & b & b\\ 0 & c & c & c\\ 0 & 0 & c & c\\ 0 & 0 & c & c\end{bmatrix}\Rightarrow\begin{bmatrix} b & b & b & b\\ 0 & c & c & c\\ 0 & 0 & c & d\\ 0 & 0 & 0 & d\end{bmatrix}=R$$

$$G_{n-1,n-1} * R_{1,n-1} = R$$

$$G_{n-1,n-1} * \ldots * G_{2,2}\ldots * G_{n-2,2} * G_{n-1,2} * G_{1,1}\ldots * G_{n-2,1} * G_{n-1,1} * A = R$$

$$A = Q * R \text{ con } \mathrm{Q}^{\mathrm{T}} = G_{n-1,n-1} * \ldots * G_{2,2}\ldots * G_{n-2,2} * G_{n-1,2} * G_{1,1}\ldots * G_{n-2,1} * G_{n-1,1}$$

Algoritmo: triangularización de Givens

Para $j = 1,2,\ldots,n$

Para $i = m, m-1, \ldots, j+1$

Calcular c y s tales que $\begin{bmatrix} c & s \\ -s & \mathrm{c} \end{bmatrix} \begin{bmatrix} a_{i-1,j} \\ a_{i,j} \end{bmatrix} = \begin{bmatrix} \sqrt{a_{i-1,j}{}^2 + a_{i,j}{}^2} \\ 0 \end{bmatrix}$

Si $\left|a_{i,j}\right| < \left|a_{i-1,j}\right|$

$t = a_{i,j} / a_{i-1,j}; c = 1/\sqrt{1+t^2}; s = c * t;$

Si no

$t = a_{i-1,j} / a_{i,j}; s = 1/\sqrt{1+t^2}; c = s * t;$

Para $k = j, j+1, \ldots, n$

$u = a_{i-1,k}; w = a_{i,k};$

$$\begin{bmatrix} a_{i-1,k} \\ a_{i,k} \end{bmatrix} = \begin{bmatrix} c * u + s * w \\ -s * u + c * w \end{bmatrix}$$

Finpara

Finpara

Finpara

Operaciones básicas con transformaciones ortogonales

Anulación ortogonal de las componentes de un vector vía Rotaciones de Givens

$$G_i = \left[\begin{array}{c|cc|c} I_{i-1} & 0 & & 0 \\ \hline 0 & c_i & s_i & 0 \\ & -s_i & c_i & \\ \hline 0 & 0 & & I_{n-i+1} \end{array}\right]$$

Dado $x \in \Re^n$, encontrar $H \in \Re^{nxn}$, tal que $H^T H = I$ y $Hx = ke_1$

Eligiendo convenientemente α_i : $$G_1 * \ldots * G_{n-2} * G_{n-1} x = k * e_1$$

Coste: 7(n-1)+6(n-1)=13(n-1)Flops

Coste total: O(n)

Reducción ortogonal de una matriz de Hessenberg superior a la forma triangular

$$H=\begin{bmatrix} h & h & h & h & h \\ h & h & h & h & h \\ & h & h & h & h \\ & & h & h & h \\ & & & h & h \end{bmatrix} \Rightarrow R=\begin{bmatrix} r & r & r & r & r \\ 0 & r & r & r & r \\ & 0 & r & r & r \\ & & 0 & r & r \\ & & & 0 & r \end{bmatrix}$$

$$Encontrar\, G_1, G_2, ..., G_{n-1} \text{ tales que}$$

$$G_{n-1} * G_{n-2} * ... * G_2 * G_1 * H = R$$

$$G_{n-1} * G_{n-2} * ... * G_1 * H = \begin{bmatrix} h & h & h & h & h \\ 1 & h & h & h & h \\ & 2 & h & h & h \\ & & 3 & h & h \\ & & & 4 & h \end{bmatrix} = R = \begin{bmatrix} r & r & r & r & r \\ 0 & r & r & r & r \\ & 0 & r & r & r \\ & & 0 & r & r \\ & & & 0 & r \end{bmatrix}$$

Coste: $7(n-1)+(6n+6(n-1)+\ldots+6*2)= (7n+(6n^2)/2)=3n^2$Flops

Coste total: $O(n^2)$

Producto por matrices de Givens

Analizar el coste de calcular : $Q * G_{n-1} * G_{n-2} * \ldots * G_2 * G_1$

con $Q \in \Re^{nxn}$

$$\begin{bmatrix} q & q & q & q \\ q & q & q & q \\ q & q & q & q \\ q & q & q & q \end{bmatrix} \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & c & s \\ & & -s & c \end{bmatrix} \begin{bmatrix} 1 & & & \\ & c & s & \\ & -s & c & \\ & & & 1 \end{bmatrix} \begin{bmatrix} c & s & & \\ -s & c & & \\ & & 1 & \\ & & & 1 \end{bmatrix}$$

$$Coste: \ 6n + 6n + \ldots (n-1) \ veces \ldots + 6n = 6n^2 \, Flops$$

Coste total: O(n²)

Actualización de la QR: Modificación de rango 1

Sea $A \in \Re^{nxn}, A = QR$, descomposición QR de A, y $u, v \in \Re^n$

Se quiere calcular Q_1, R_1 tales que $A + uv^T = Q_1 R_1$, descomposición QR de $A + uv^T$

$$\text{Sea } w = Q^T u, \text{ y } P = G_1 * G_2 * \ldots * G_{n-1} \,/\, Pw = ke_1$$

Coste: $2n^2$ Flops

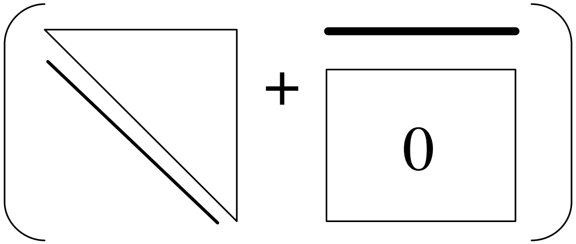
Coste: 13n Flops

$$\text{Se tiene } A + uv^T = QR + uv^T = Q(R + Q^T uv^T) = Q(R + wv^T) =$$

$$QP^T P(R + wv^T) = QP^T(PR + ke_1 v^T) = QP^T * H$$

con $H = PR + ke_1 v^T$ Hessenberg superior

Coste: $O(n^2)$



Eligiendo G ortogonal, tal que $GH = R_1$ triangular superior

Coste: $3n^2$ Flops

$$A + uv^T = QP^T G^T GH = Q_1 R_1$$

siendo $Q_1 = QP^T G^T$ Coste: $O(n^2)$

Coste total: $O(n^2)$

Actualización de la QR: Si se quita una columna

Sea $A \in \Re^{mxn}, A = QR$, descomposición QR de A

Se quiere calcular

$$Q_1, R_1 \text{ tales que } A_1 = [a_1, a_2, \ldots, a_{k-1}, a_{k+1}, \ldots, a_n] = Q_1 R_1, \text{ descomposición } QR \text{ de } A_1$$

$$\text{Se tiene que } Q^T A_1 = [r_1, r_2, \ldots, r_{k-1}, r_{k+1}, \ldots, r_n] = H = \begin{bmatrix} r & r & r & r & r \\ & r & r & r & r \\ & & r & r & r \\ & & & r & r \\ & & & r & r \\ & & & & r \end{bmatrix}$$

H es Hessenberg superior

Eligiendo G ortogonal, tal que $G\mathrm{H} = R_1$ triangular superior Coste: $3n^2$ Flops

se tiene $Q^T A_1 = G^T G H = G^T R_1 \Rightarrow A_1 = Q G^T R_1 = Q_1 R_1$

con $Q G^T = Q_1$

Coste: $O(n^2)$

Coste total: $O(n^2)$

Actualización de la QR: Si se añade una columna

Sea $A \in \Re^{mxn}, A = QR,$ descomposición QR de A

e quiere calcular Q_1, R_1 tales que $A_1 = [a_1, a_2, \ldots, a_{k-1}, z, a_{k+1}, \ldots, a_n] = Q_1 R_1,$ desc. QR de A_1

$$Se \text{ tiene que } Q^T A_1 = [r_1, r_2, \ldots, r_{k-1}, Q^T z, r_{k+1}, \ldots, r_n] = \begin{bmatrix} r & r & r & s & r & r \\ & r & r & s & r & r \\ & & r & s & r & r \\ & & & s & r & r \\ & & & s & & r \\ & & & s & & \\ & & & s & & \end{bmatrix}$$

Coste: O(m²)

$$\text{Eligiendo } G \text{ ortogonal, tal que } G(Q^T z) = \begin{bmatrix} \bar{z}_1 \\ \bar{z}_2 \\ . \\ \bar{z}_{k+1} \\ 0 \\ . \\ 0 \end{bmatrix} = \bar{z}$$

Coste: O(m) Flops

$$\text{se tiene } Q^T A_1 = G^T G[r_1, r_1, \ldots r_{k-1}, Q^T z, r_{k+1}, \ldots, r_n] = G^T [r_1, r_1, \ldots r_{k-1}, \bar{z}, r_{k+1}, \ldots, r_n] = G^T R_1$$

$$\Rightarrow A_1 = QG^T R_1 = Q_1 R_1$$

con $QG^T = Q_1$ Coste: O(m²)

Coste total: O(m²)

Actualización de la QR: Si se añade una fila

Sea $A \in \Re^{mxn}$, $A = QR$, descomposición QR de A

Se quiere calcular Q_1, R_1 tales que $A_1 = \begin{bmatrix} z^T \\ A \end{bmatrix} = Q_1 R_1$, (desc. QR de A_1)

Se tiene que $\begin{bmatrix} z^T \\ A \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & Q \end{bmatrix} \begin{bmatrix} z^T \\ R \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & Q \end{bmatrix} H$, con $H = \begin{bmatrix} z^T \\ R \end{bmatrix}$ Hessenberg superior

$$H = \begin{bmatrix} z^T \\ R \end{bmatrix} = \begin{bmatrix} z & z & z & z \\ r & r & r & r \\ & r & r & r \\ & & r & r \\ & & & r \end{bmatrix}$$

Si se elige G ortogonal, tal que $GH = R_1$ se tiene $\begin{bmatrix} z^T \\ A \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & Q \end{bmatrix} G^T G H = Q_1 R_1$

Coste: $3n^2$ Flops

con $Q_1 = \begin{bmatrix} 1 & 0 \\ 0 & Q \end{bmatrix} G^T$

Coste: $O(m^2)$ Flops

Coste total: $O(m^2)$

Actualización de la QR: Si se añade una fila (caso general)

Sea $A \in \Re^{mxn}, A = QR,$ descomposición QR de A

Se quiere calcular

$$Q_2, R_2 \text{ tales que } A_2 = \begin{bmatrix} A_{(1)} \\ z^T \\ A_{(2)} \end{bmatrix} = Q_2 R_2, \text{ (desc. } QR \text{ de } A_2)$$

Sea $\prod$ una matriz de permutación tal que $\prod A_2 = A_1 = \begin{bmatrix} z^T \\ A \end{bmatrix} = Q_1 R_1, \text{ (desc. } QR \text{ de } A_1)$

$$A_2 = \prod^T Q_1 R_1 = Q_2 R_2, \text{ (desc. } QR \text{ de } A_2)$$

$$\text{con } Q_2 = \prod^T Q_1 \text{ y } R_2 = R_1$$

Coste total: O(m²)

Actualización de la QR: Si se quita una fila

Coste: $3n^2$ Flops

Sea $A \in \Re^{mxn}, A = \begin{bmatrix} z^T \\ A_1 \end{bmatrix} = QR$, descomposición QR de A

Se quiere calcular Q_1, R_1 tales que $A_1 = Q_1 R_1$, (desc. QR de A_1)

Sea $Q = \begin{bmatrix} q_1^T \\ q_2^T \\ .. \\ q_1^T \end{bmatrix}$; Buscar un G ortogonal tal que $Gq_1 = ke_1, k = \pm 1$ Coste: 13m Flops

Se tiene $QG^T = \begin{bmatrix} k & 0 \\ 0 & Q_1 \end{bmatrix}$ Coste: O(m²) Flops

ya que $q_1^T G^T = (Gq_1)^T = (ke_1)^T = \begin{bmatrix} k & 0 & 0 & \ldots & 0 & 0 \end{bmatrix}$

y QG^T es ortogonal, luego $\overline{Q} = QG^T$ tiene sus filas ortogonales :

$$\overline{q}_{(1)}{}^T \overline{q}_{(j)} = 0, j = 2,3,\ldots,m \Rightarrow k\overline{q}_{j1} + 0^T \begin{bmatrix} \overline{q}_{j2} \\ \\ \overline{q}_{jm} \end{bmatrix} = 0 \Rightarrow \overline{q}_{j1} = 0$$

Actualización de la QR: Si se quita una fila

Además GR es Hessenberg superior,

$$\begin{bmatrix} u & u & u & u \\ x & x & x & x \\ & x & x & x \\ & & x & x \\ & & & x \end{bmatrix}$$

Coste: O(n²) Flops

Luego

$$\begin{bmatrix} z^T \\ A_1 \end{bmatrix} = QR = QG^T GR = \begin{bmatrix} k & 0 \\ 0 & Q_1 \end{bmatrix} \begin{bmatrix} u^T \\ R_1 \end{bmatrix}, \Rightarrow A_1 = Q_1 R_1$$

Coste total: O(m²)

Actualización de la QR: Si se quita una fila (caso general)

Sea $A \in \Re^{mxn}, A = QR$, descomposición QR de A

$$\text{Supongamos } A = \begin{bmatrix} A_{(1)} \\ z^T \\ A_{(2)} \end{bmatrix} = QR, \text{ (desc. } QR \text{ de } A)$$

Sea $\prod$ una matriz de permutación tal que $\prod A = A_2 = \begin{bmatrix} z^T \\ A_1 \end{bmatrix} = \prod QR = Q_2 R$,

l problema se transforma en

$$\text{Dada } A_2 \in \Re^{mxn}, \text{con } A_2 = \begin{bmatrix} z^T \\ A_1 \end{bmatrix} = Q_2 R, \text{ descomposición } QR \text{ de } A_2$$

$$\text{calcular } Q_1, R_1 \text{ tales que } A_1 = Q_1 R_1, \text{ (desc. } QR \text{ de } A_1)$$

Coste total: O(m²)

Descomposición QR para matrices de rango deficiente

- El algoritmo de Householder falla si las matrices son de rango deficiente: rank(A)=r<n, $A \in \Re^{mxn}$ con m>=n
- Es necesario reordenar las columnas para evitar las divisiones por cero
- Una buena estrategia es anular primero aquella columna con mayor 2-norma
- Debe evitarse el cálculo de las 2-normas de todas las columnas en cada etapa

Algorithm 5.4.1 (Householder QR With Column Pivoting) Given $A \in \mathbb{R}^{m \times n}$ with $m \geq n$, the following algorithm computes $r = \text{rank}(A)$ and the factorization (5.4.1) with $Q = H_1 \cdots H_r$ and $\Pi = \Pi_1 \cdots \Pi_r$. The upper triangular part of A is overwritten by the upper triangular part of R and components $j+1{:}m$ of the jth Householder vector are stored in $A(j+1{:}m, j)$. The permutation Π is encoded in an integer vector piv. In particular, Π_j is the identity with rows j and $piv(j)$ interchanged.

```
for j = 1:n
    c(j) = A(1:m, j)^T A(1:m, j)
end
r = 0; τ = max{c(j), ... c(n)}
Find smallest k with 1 ≤ k ≤ n so c(k) = τ
while τ > 0
    r = r + 1
    piv(r) = k; A(1:m, r) ↔ A(1:m, k); c(r) ↔ c(k)
    v(r:m) = house(A(r:m, r))
    A(r:m, r:n) = row.house(A(r:m, r:n), v(r:m))
    A(r + 1:m, r) = v(r + 1:m)
    for i = r + 1:n
        c(i) = c(i) − A(r, i)^2
    end
    if r < n
        τ = max{c(r + 1), ... c(n)}
        Find smallest k with r + 1 ≤ k ≤ n so c(k) = τ.
    else
        τ = 0
    end
end
```

```
for j = 1:n
    c(j) = A(1:m, j)^T A(1:m, j)
end
r = 0; τ = max{c(1), ..., c(n)}
Find smallest k with 1 ≤ k ≤ n so c(k) = τ
while τ > 0
    r = r + 1
    piv(r) = k; A(1:m, r) ↔ A(1:m, k); c(r) ↔ c(k)
    Norm(r)
    for k=r : n
        Fact(r,k)
        Cmod(r,k)
    end
    for i = r + 1:n
        c(i) = c(i) − A(r, i)^2
    end
    if r < n
        τ = max{c(r + 1), ..., c(n)}
        Find smallest k with r + 1 ≤ k ≤ n so c(k) = τ
    else
        τ = 0
    end
end
```

Problema Lineal de Mínimos Cuadrados con matrices de rango deficiente

$$\left\|Ax-b\right\|_2^2 = \left\|Q^T A \Pi \Pi^T x - Q^T b\right\|_2^2 = \left\| \begin{bmatrix} R_{11} & R_{12} \\ 0 & 0 \end{bmatrix} y - \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} \right\|_2^2$$

$$= \left\|R_{11} y_1 + R_{12} y_2 - c_1\right\|_2^2 + \left\|- c_2\right\|_2^2$$

QR con pivotamiento: $A\Pi = QR \Leftrightarrow A = QR\Pi^T$

$R_{11} y_1 = c_1 - R_{12} y_2 \quad \Rightarrow$ Infinitas soluciones

Posible Solución: $\quad y_2 = 0; \qquad y_1 = R_{11}^{-1} c_{12}$

Re*siduo*: $\left\|c_2\right\|_2$

Problema Lineal de Mínimos Cuadrados con matrices de rango deficiente:

Solución de norma mínima

$$\text{QR con pivotamiento}: \; A\Pi = QR \Leftrightarrow A = QR\Pi^T \quad \text{con} \;\; R = \begin{bmatrix} R_{11} & R_{12} \\ 0 & 0 \end{bmatrix}$$

$$\text{Calculamos } Z, nxn, ortogonal, \text{ tal que } Z^T \begin{bmatrix} R_{11}{}^T \\ R_{12}{}^T \end{bmatrix} = \begin{bmatrix} T_{11}{}^T \\ 0 \end{bmatrix},$$

$$\text{con } T_{11}{}^T \text{ triangular superior. Se tiene} \quad Q^T A\Pi Z = \begin{bmatrix} R_{11} & R_{12} \\ 0 & 0 \end{bmatrix} Z = \begin{bmatrix} T_{11} & 0 \\ 0 & 0 \end{bmatrix}$$

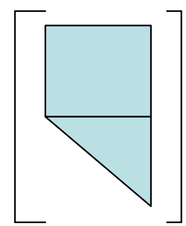
$$\|Ax - b\|_2^2 = \|Q^T A\Pi Z Z^T \Pi^T x - Q^T b\|_2^2 = \left\| \begin{bmatrix} T_{11} & 0 \\ 0 & 0 \end{bmatrix} y - \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} \right\|_2^2, \text{con } y = Z^T \Pi^T x$$

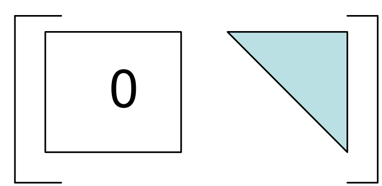
$$= \|T_{11} y_1 - c_1\|_2^2 + \|- c_2\|_2^2 \qquad \text{El primer sumando se anula si} \quad T_{11} y_1 = c_1$$

$$\textit{Solución de norma mínima}: \; y_2 = 0; \; y_1 = T_{11}{}^{-1} c_{12}; \; \text{Residuo}: \; \|c_2\|_2 \; ; x = \Pi Z y$$

Extensiones de la Descomposición QR: Factorización RQ

Dada una matriz $A \in \Re^{mxn}$ descomponerla en un producto de matrices A=RQ, donde $Q \in \Re^{nxn}$ es ortogonal y $R \in \Re^{mxn}$ es triangular superior con respecto a una diagonal secundaria

$AQ^T = R =$  $= \begin{bmatrix} R_1 \\ R_2 \end{bmatrix}$

$AQ^T = R =$ 

$= \begin{bmatrix} 0 & R_2 \end{bmatrix}$

Extensiones de la Descomposición QR: Factorización QR generalizada (QRG)

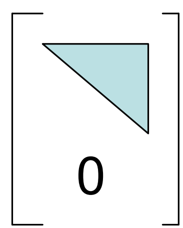
Dada dos matrices $A \in \Re^{mxn}$ y $B \in \Re^{mxp}$ descomponerlas en un productos

A=QR, y B=QSV^T donde $Q \in \Re^{mxm}$ y $V \in \Re^{pxp}$ son ortogonales y

$R \in \Re^{mxn}$ es triangular superior con respecto a la diagonal principal, y

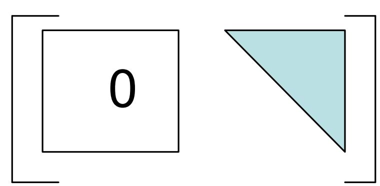
$S \in \Re^{mxp}$ es triangular superior con respecto a una diagonal secundaria

$Q^T A = R \quad (m \geq n) \quad =$



$$= \begin{array}{c c} \begin{bmatrix} R_1 \\ 0 \end{bmatrix} & \begin{array}{c} n \\ m\text{-}n \end{array} \\ n & \end{array}$$

$Q^T BV = S \quad (m < p) \quad =$



$$= \begin{array}{c c} \begin{bmatrix} 0 & S_2 \end{bmatrix} & m \\ \begin{array}{cc} p\text{-}m & m \end{array} & \end{array}$$

Extensiones de la Descomposición QR: Factorización QR generalizada (QRG)

Dada dos matrices $A \in \Re^{mxn}$ y $B \in \Re^{mxp}$ descomponerlas en un productos

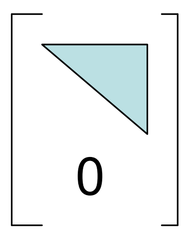
A=QR, y B=QSV^T donde $Q \in \Re^{mxm}$ y $V \in \Re^{pxp}$ son ortogonales y

$R \in \Re^{mxn}$ es triangular superior con respecto a la diagonal principal, y

$S \in \Re^{mxp}$ es triangular superior con respecto a una diagonal secundaria

$Q^TA = R =$

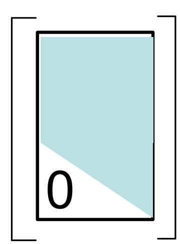
$(m \geq n)$



$$= \begin{array}{cc} \begin{bmatrix} R_1 \\ 0 \end{bmatrix} & \begin{array}{c} n \\ m-n \end{array} \\ n & \end{array}$$

$Q^T BV = S$ =

$(m > p)$

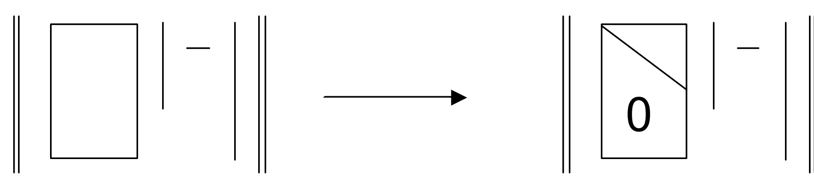


$$= \begin{array}{cc} \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} & \begin{array}{c} m-p \\ p \end{array} \\ p & \end{array}$$

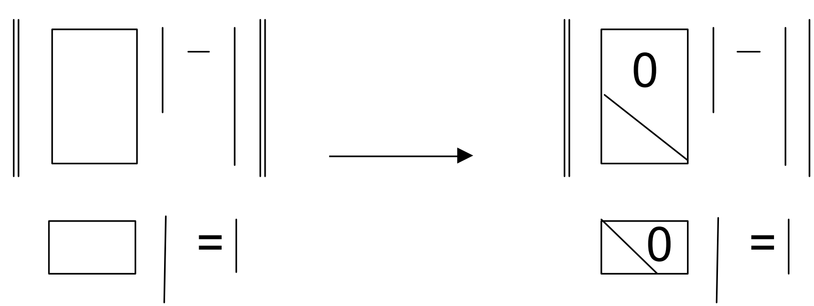
Resolución del Problema de Mínimos Cuadrados con Restricciones de Igualdad

$$\min_{Bx=d} \left\| Ax - b \right\|_2$$

Técnica de triangularización en PLMC sin restricciones



Técnica de triangularización en PLMC con restricciones



$$\min_{Bx=d} \left\| Ax - b \right\|_2$$

$$A \in \Re^{mxn}, m \geq n, rank(A) = n$$

$$B \in \Re^{pxn}, p \leq n, rank(B) = p$$

$$rank\left(\begin{bmatrix} A \\ B \end{bmatrix}\right) = n$$

Calculemos la Descomposición GQR de B^T y A^T

$$Q^T B^T = S = \begin{array}{c} \begin{bmatrix} S_{11} \\ 0 \end{bmatrix} \end{array} \begin{array}{l} p \\ n-p \end{array}$$

$$Q^T A^T U = R = \begin{array}{ccc|l} & & & \\ \end{array} \begin{array}{c} \begin{bmatrix} 0 & R_{11} & R_{12} \\ 0 & 0 & R_{22} \end{bmatrix} \\ \begin{array}{ccc} m-n & \quad p \quad & n-p \end{array} \end{array} \begin{array}{l} p \\ n-p \\ \ \end{array}$$

$$Sea \quad Q = \underset{\begin{array}{cc} n & n-p \end{array}}{[Q_1 \quad Q_2]}\ n \quad \text{y} \quad U = \underset{\begin{array}{ccc} m\text{-}n & p & n\text{-}p \end{array}}{[U_1 \quad U_2 \quad U_3]}\ m$$

$$\text{Es decir} \quad Q^T = \underset{n}{\begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix}} \begin{array}{l} p \\ n\text{-}p \end{array} \quad \text{y} \quad U^T = \underset{m}{\begin{bmatrix} U_1^T \\ U_2^T \\ U_3^T \end{bmatrix}} \begin{array}{l} m\text{-}n \\ p \\ n\text{-}p \end{array}$$

Si llamamos $y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = Q^T x = \begin{array}{c} \begin{bmatrix} Q_1^T x \\ Q_2^T x \end{bmatrix} \end{array} \begin{array}{c} p \\ n\text{-}p \end{array}$ y $\mathrm{c} = \begin{bmatrix} \mathrm{c}_1 \\ \mathrm{c}_2 \\ \mathrm{c}_3 \end{bmatrix} = U^T b = \begin{bmatrix} U_1^T b \\ U_2^T b \\ U_3^T b \end{bmatrix} \begin{array}{c} m\text{-}n \\ p \\ n\text{-}p \end{array}$

Podemos resolver el problema según :

$$Q^T A^T U = R \Rightarrow A^T = QRU^T \Rightarrow A = UR^T Q^T$$

$$Q^T B^T = S \Rightarrow B = S^T Q^T$$

$$A \to \begin{array}{cc} \begin{bmatrix} 0 & 0 \\ R_{11}^T & 0 \\ R_{12}^T & R_{22}^T \end{bmatrix} & \begin{array}{c} m-n \\ p \\ n-p \end{array} \\ \begin{array}{cc} p & \quad n-p \end{array} & \end{array}$$

$$B \to \begin{array}{c} \begin{bmatrix} S_{11}^T & 0 \end{bmatrix} \\ \begin{array}{cc} p & n-p \end{array} \end{array}$$

y

$$\|Ax - b\|_2 = \|UR^T Q^T x - b\|_2 = \|U^T (UR^T Q^T x - b)\|_2 = \|R^T Q^T x - U^T b\|_2 = \|R^T y - c\|_2$$

Es decir $\|Ax - b\|_2 = \left\| \begin{bmatrix} 0 & 0 \\ R_{11}^T & 0 \\ R_{12}^T & R_{22}^T \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} - \begin{bmatrix} \mathrm{c}_1 \\ \mathrm{c}_2 \\ \mathrm{c}_3 \end{bmatrix} \right\|_2$ (1)

con la condición $Bx = d \Rightarrow S^T Q^T x = d \Rightarrow S^T y = d \Rightarrow \begin{array}{c} \begin{bmatrix} S_{11}^T & 0 \end{bmatrix} \\ \begin{array}{cc} p & n\text{-}p \end{array} \end{array} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = d$

De aquí se tiene $S_{11}{}^{T} y_1 = d \Rightarrow y_1 = S_{11}{}^{-T} d$

Sustituyendo en (1) el problema se convierte en calcular

$$\min \left\| R_{22}{}^{T} y_2 - (c_3 - R_{12}{}^{T} y_1) \right\|_2$$

cuya solución es $y_2 = R_{22}{}^{-T}(c_3 - R_{12}{}^{T} S_{11}{}^{-T} d)$

De aquí se tiene que el mínimo se alcanza en

$$x = Qy = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = Q_1 y_1 + Q_2 y_2$$

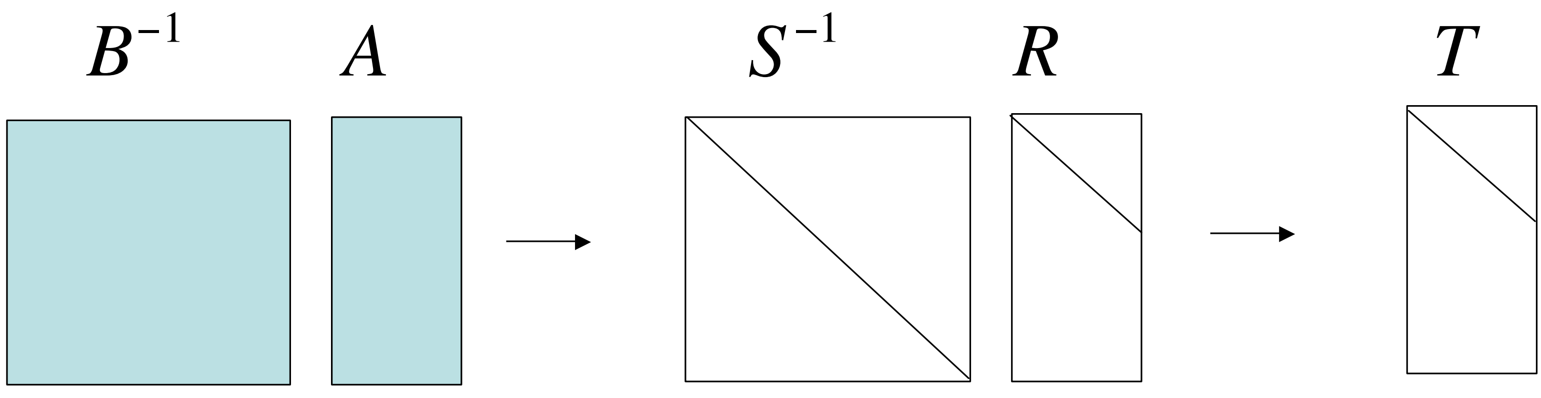
y toma el valor ρ tal que

$$\rho^2 = \left\| c_1 \right\|_2^2 + \left\| R_{11}{}^{T} y_1 - c_2 \right\|_2^2$$

Problema de mínimos cuadrados ponderado

$$\min_{x} \left\| B^{-1}(Ax - b) \right\|_2 \quad , A \in \Re^{mxn}, B \in \Re^{mxm}$$

Técnica de triangularización en PMC ponderado



$Si\ conocemos\ A = QR \quad y\ Q^T BV = S$

$se\ tiene\ que$

$$V^T(B^{-1}A) = \begin{bmatrix} T \\ 0 \end{bmatrix} \quad y \quad V^T B^{-1} = S^{-1}Q^T$$

Por tanto

$$\left\| B^{-1}(Ax-b) \right\|_2 = \left\| V^T B^{-1}(Ax-b) \right\|_2 = \left\| \begin{bmatrix} T \\ 0 \end{bmatrix} x - V^T B^{-1} b \right\|_2 = \left\| \begin{bmatrix} T \\ 0 \end{bmatrix} x - \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} \right\|_2$$

$$con\ \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = c = V^T B^{-1} b \Rightarrow c = S^{-1}Q^T b = \begin{array}{c|l} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} & \begin{array}{l} n \\ m-n \end{array} \end{array}$$

$o\ bien\ Sc = Q^T b$

$Luego\ \min \left\| B^{-1}(Ax-b) \right\|_2 = \left\| c_2 \right\|_2$

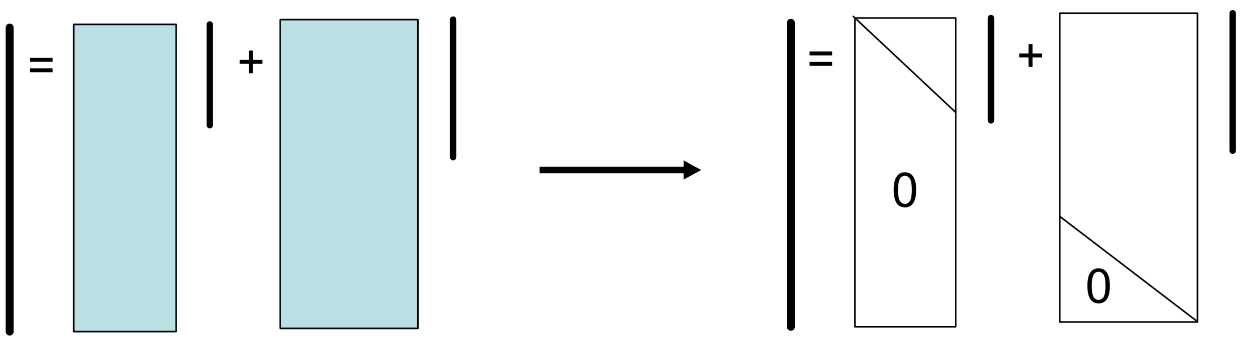
$y\ se\ obtiene\ x_0 = T^{-1} c_1$

Problema Generalizado de Mínimos Cuadrados

$$\min u^T u \qquad A \in \Re^{mxn},\ m \geq n \qquad B \in \Re^{mxp}, m \geq p$$

$$b = Ax + Bu \qquad rank(A) = n, \qquad rank(B) = p$$

Técnica de triangularización en PGMC



Calculemos la Descomposición GQR de A y B

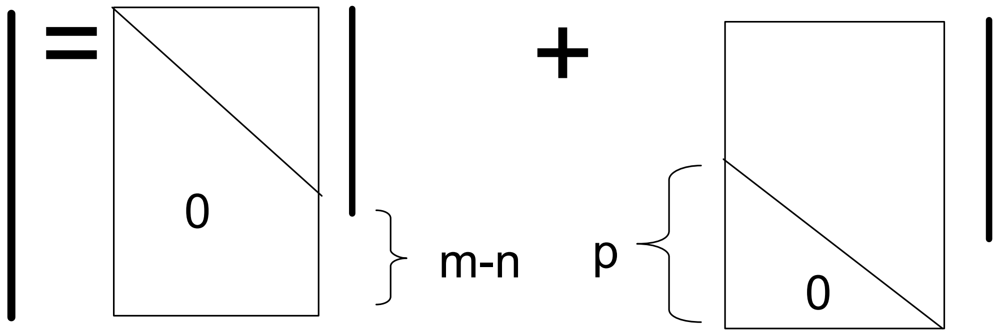
$$Q^T A = R = \begin{array}{cc} \begin{bmatrix} R_1 \\ 0 \end{bmatrix} & \begin{array}{c} n \\ m-n \end{array} \\ n & \end{array} \qquad Q^T BV = S = \begin{array}{cc} \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} & \begin{array}{c} m-p \\ p \end{array} \\ p & \end{array}$$

$$Q^T b = Q^T Ax + Q^T BVV^T u \Rightarrow \quad c = Rx + Sv$$

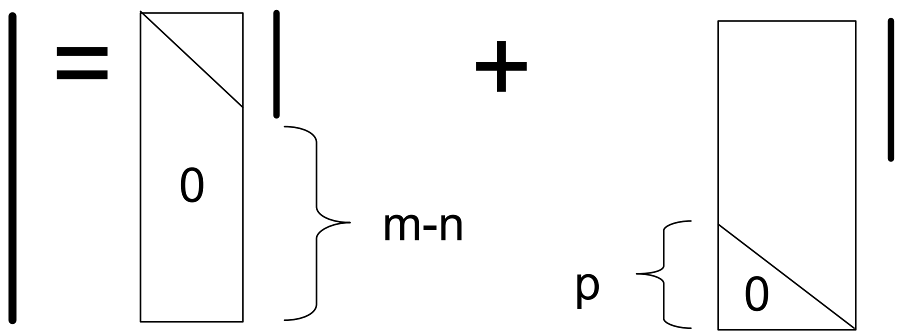
$$con \quad c = Q^T b, \qquad v = V^T u \Rightarrow u = Vv$$

$$\begin{array}{ccccc} \min\ u^T u & = & \min\ (Vv)^T (Vv) & = & \min\ v^T v \\ b = Ax + Bu & & c = Rx + Sv & & c = Rx + Sv \end{array}$$

Casos a estudiar según el valor de p



p>=m-n



p<m-n

p>m-n

A → R

m=10	n=6	m-n=4					
r	r	r	r	r	r		x
0	r	r	r	r	r		x
0	0	r	r	r	r		x
0	0	0	r	r	r		x
0	0	0	0	r	r		x
0	0	0	0	0	r		x
0	0	0	0	0	0		
0	0	0	0	0	0		
0	0	0	0	0	0		
0	0	0	0	0	0		
p=7							v
s	s	s	s	s	s	s	v1
s	s	s	s	s	s	s	v1
s	s	s	s	s	s	s	v1
s	s	s	s	s	s	s	v2
0	s	s	s	s	s	s	v2
0	0	s	s	s	s	s	v2
0	0	0	s	s	s	s	v2
0	0	0	0	s	s	s	
0	0	0	0	0	s	s	
0	0	0	0	0	0	s	

B → S

Caso p ≥ m-n

$$c = Rx + Sv \Rightarrow \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} R_1 \\ 0 \end{bmatrix} x + \begin{bmatrix} S_{11} & S_{12} \\ 0 & S_{22} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

$$con \quad v = \begin{array}{cl} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} & \begin{array}{c} p-m+n \\ m-n \end{array} \end{array} \qquad y \qquad S = \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} = \begin{array}{cc} \begin{bmatrix} S_{11} & S_{12} \\ 0 & S_{22} \end{bmatrix} & \begin{array}{c} n \\ m-n \end{array} \\ \begin{array}{cc} p-m+n & m-n \end{array} & \end{array}$$

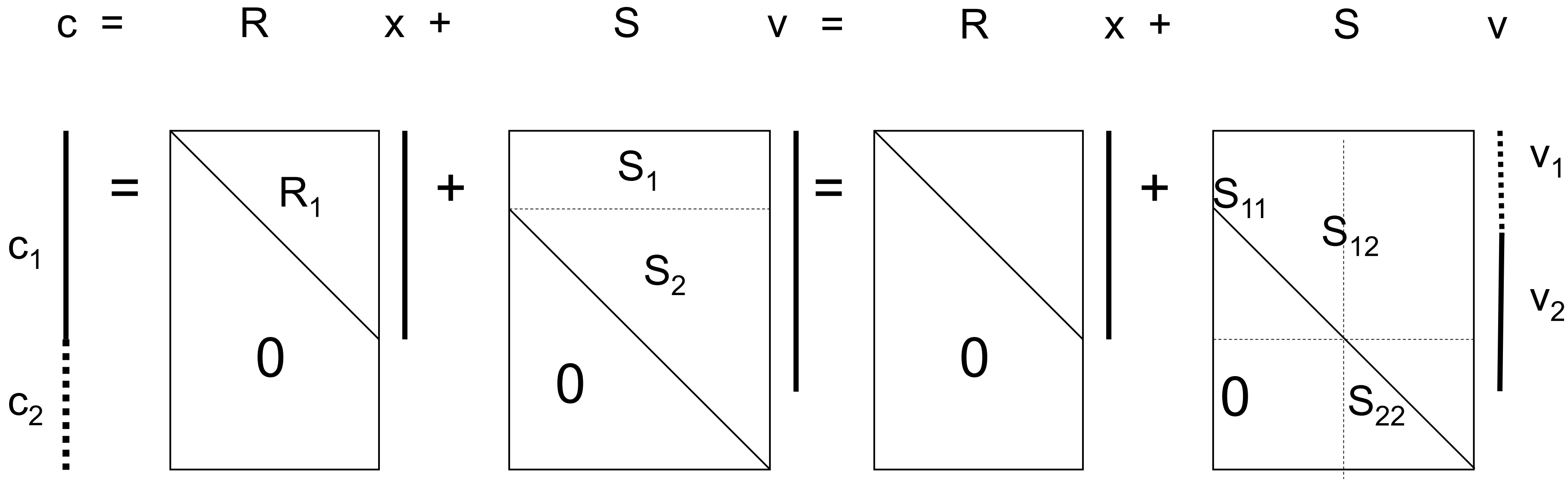
$$c_1 = R_1 x + S_{11} v_1 + S_{12} v_2$$

$$c_2 = S_{22} v_2 \Rightarrow v_2 = S_{22}^{-1} c_2$$

$$v_1 = 0 \qquad \textit{(para que la norma de v sea mínima)}$$

$$x = R_1^{-1}(c_1 - S_{12} v_2)$$

$$p \geq m - n$$



$p<m-n$

m=10	n=6	m-n=4					
r	**r**	**r**	**r**	**r**	**r**		
0	**r**	**r**	**r**	**r**	**r**		
0	**0**	**r**	**r**	**r**	**r**		
0	**0**	**0**	**r**	**r**	**r**		
0	**0**	**0**	**0**	**r**	**r**		
0	**0**	**0**	**0**	**0**	**r**		
0	0	0	0	0	0		
0	0	0	0	0	0		
0	0	0	0	0	0		
0	0	0	0	0	0		
p=3			v				
s	**s**	**s**	**v**				
s	**s**	**s**	**v**				
s	**s**	**s**	**v**				
s	**s**	**s**					
s	**s**	**s**					
s	**s**	**s**					
s	**s**	**s**					
s	**s**	**s**					
0	**s**	**s**					
0	**0**	**s**					

Caso p < m-n

$$c = Rx + Sv \Rightarrow c = \begin{bmatrix} R_1 \\ 0 \end{bmatrix} x + \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} v$$

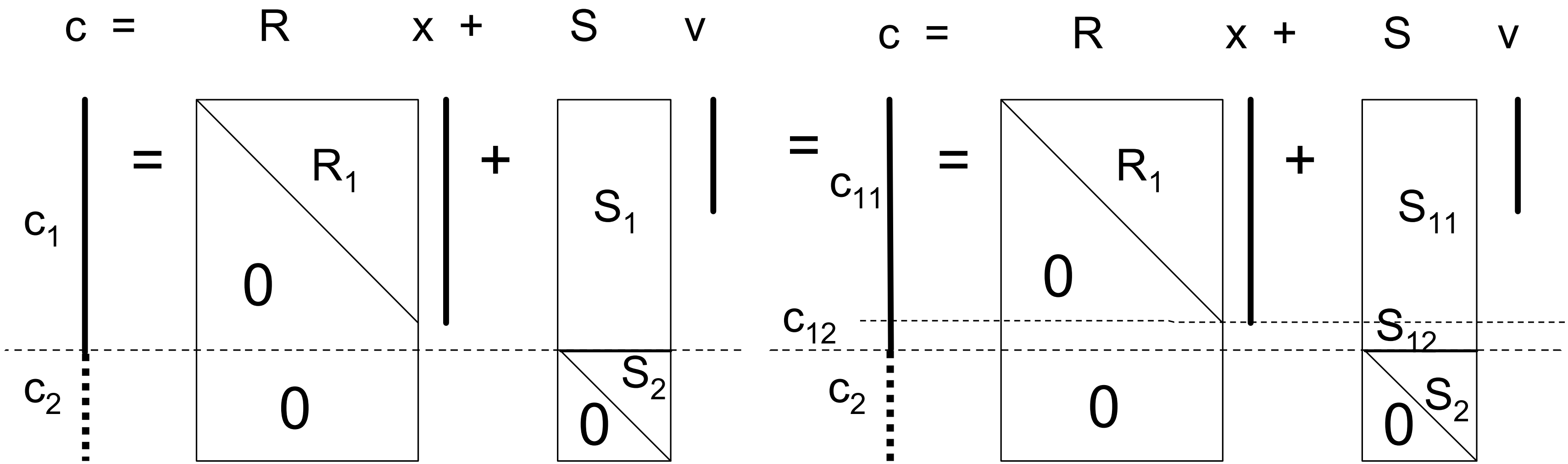
$$\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} R_1^{'} \\ 0 \end{bmatrix} x + \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} v \qquad \text{con } R_1^{'} = \begin{array}{cc} \begin{bmatrix} R_1 \\ 0 \end{bmatrix} & \begin{array}{c} n \\ m-n-p \end{array} \end{array} \qquad \text{y S} = \begin{array}{cc} \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} & \begin{array}{c} m-p \\ p \end{array} \end{array}$$

$$c_2 = S_2 v \Rightarrow v = S_2^{-1} c_2$$

$$c_1 = \begin{bmatrix} c_{11} \\ c_{12} \end{bmatrix} = \begin{bmatrix} R_1 \\ 0 \end{bmatrix} x + S_1 v = \begin{bmatrix} R_1 \\ 0 \end{bmatrix} x + \begin{bmatrix} S_{11} \\ S_{12} \end{bmatrix} v \Rightarrow$$

$$\Rightarrow c_{11} = R_1 x + S_{11} v \Rightarrow x = R_1^{-1}(c_{11} - S_{11} v)$$

$p<m-n$



Problema lineal de mínimos cuadrados discreto

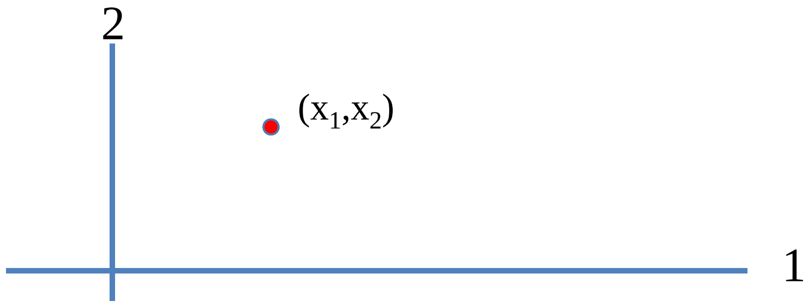
Encontrar $x_{ML} \in K \subset A^n$ tal que $\left\| Hx_{ML} - b \right\|_2 = \min\limits_{x \in K} \left\| Hx - b \right\|_2$

con H $\in \Re^{m \times n}$, $A = \{\alpha_1, \alpha_2, ..., \alpha_t\}$ y $t \in$ N

Problema lineal de mínimos cuadrados discreto

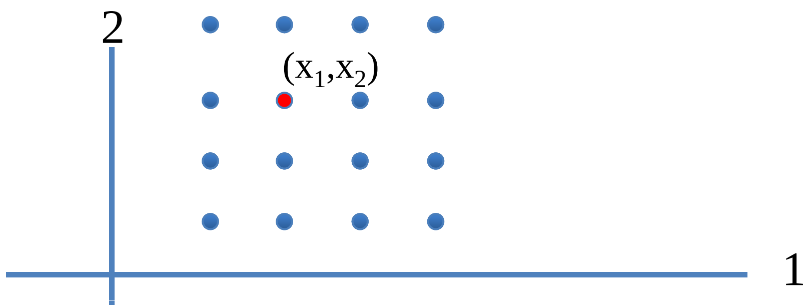
Problema continuo

$$\text{Encontrar } x_{LS} \in R^n \ \text{ tal que } \left\| Hx_{LS} - b \right\|_2 = \min_{x \in R^n} \left\| Hx - b \right\|_2$$



Problema discreto

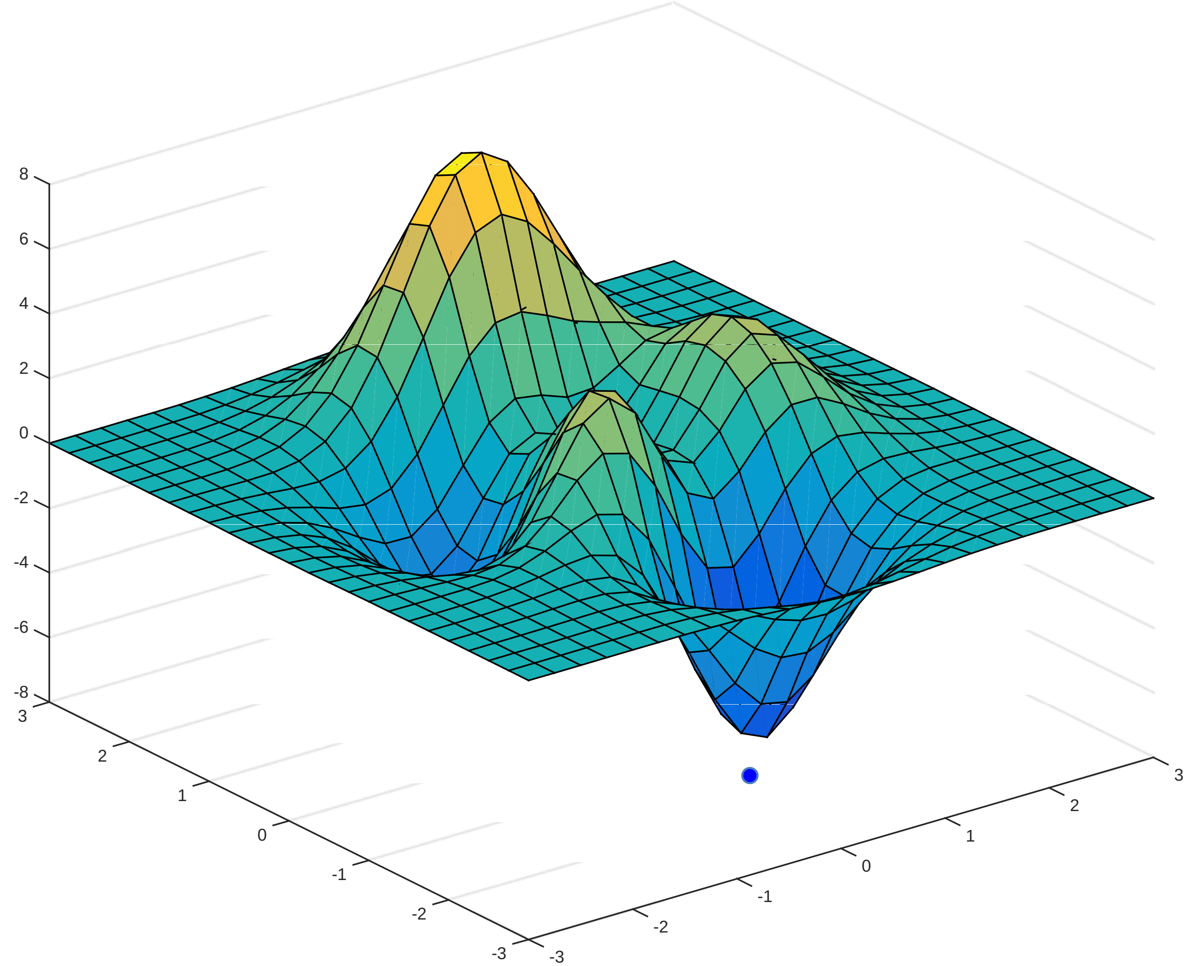
$$\text{Encontrar } x_{ML} \in K \subset A^n \ \text{ tal que } \left\| Hx_{ML} - b \right\|_2 = \min_{x \in K} \left\| Hx - b \right\|_2$$



Problema lineal de mínimos cuadrados discreto

Problema continuo

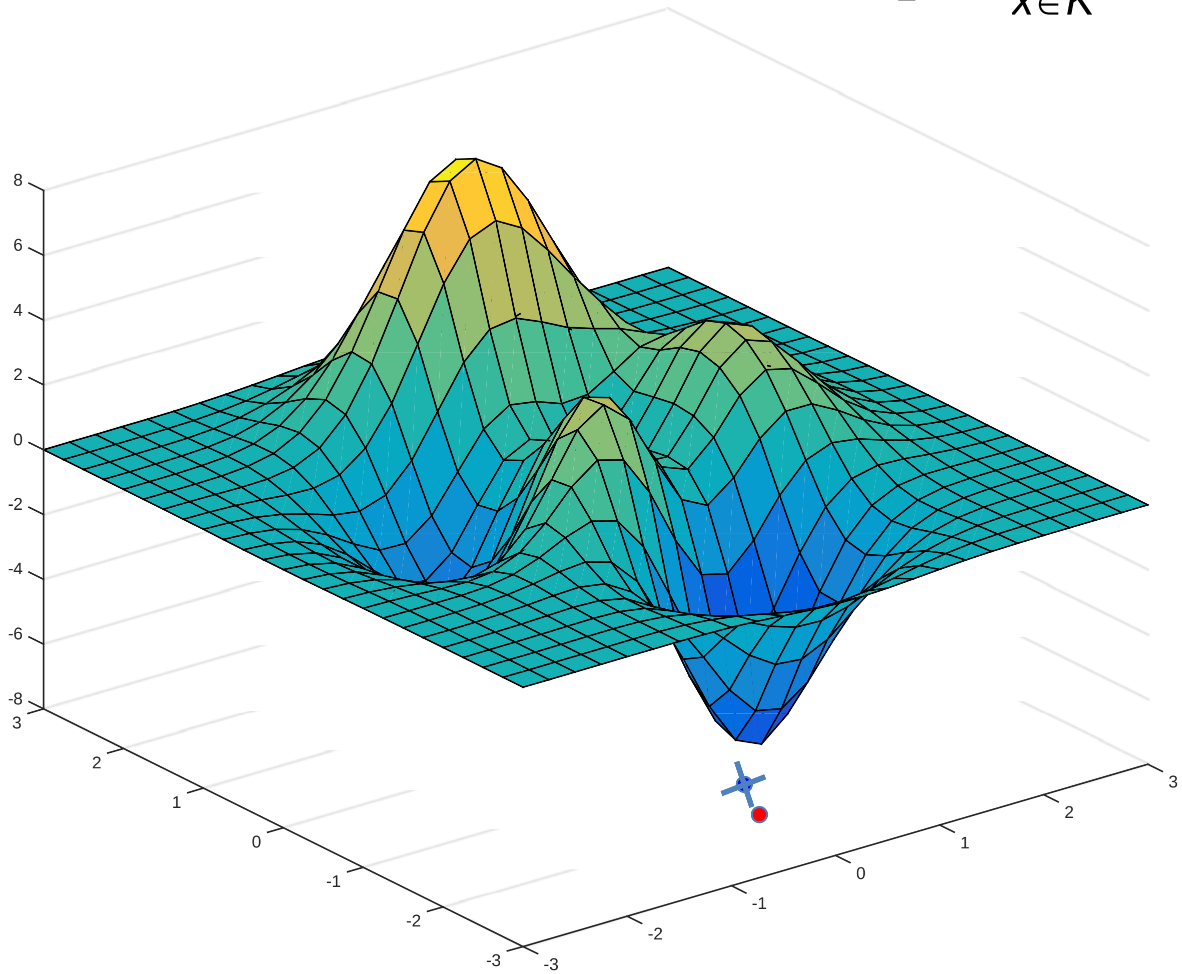
$$\text{Encontrar } x_{LS} \in R^n \text{ tal que } \left\| Hx_{LS} - b \right\|_2 = \min_{x \in R^n} \left\| Hx - b \right\|_2$$



Problema lineal de mínimos cuadrados discreto

Problema discreto

Encontrar $x_{ML} \in K \subset A^n$ tal que $\left\| H x_{ML} - b \right\|_2 = \min_{x \in K} \left\| H x - b \right\|_2$



Problema lineal de mínimos cuadrados discreto

Algoritmos de resolución:

Algoritmos ML (óptimos): algoritmos de máxima probabilidad (maximum likelihood) que encuentran la solución exacta

Algoritmos subóptimos: encuentran una solución aproximada

Algoritmos de resolución:

1. Búsqueda exhaustiva (ML)
2. Zero forcing (subóptimo)
3. Sphere decoding (ML)
4. Variantes óptimas y subóptimas del sphere decoding

Problema lineal de mínimos cuadrados discreto

$$\text{Encontrar } x_{ML} \in K \subset A^n \text{ tal que } \|Hx_{ML} - b\|_2 = \min_{x \in K} \|Hx - b\|_2$$

$$\text{siendo } A = \{\alpha_1, \alpha_2, ..., \alpha_t\} \text{ y } t \in \mathrm{N}$$

Caso de estudio concreto (constelación L-PAM)

$$\text{Encontrar } x_{ML} \in A^4 \text{ tal que } \|Hx_{ML} - b\|_2 = \min_{x \in A^4} \|Hx - b\|_2$$

$$\text{siendo } H \in R^{4\times 4},\ t \in \{2,4,8,16,32,64,128,256\},\ L = t, \alpha = \frac{-t+1}{2},\ A = \{\alpha + k, k = 0,1,2,...,t\text{-}1\}$$

Ejemplo:

$$\text{Encontrar } x_{ML} \in A^4 \text{ tal que } \|Hx_{ML} - b\|_2 = \min_{x \in A^4} \|Hx - b\|_2$$

$$\text{siendo } H = \begin{bmatrix} 1 & 0.2 & 0.3 & 0.4 \\ 0.2 & 2 & 0.5 & 0.1 \\ 0.1 & 0.9 & 3 & 0.2 \\ 0.6 & 0.7 & 0.3 & 4 \end{bmatrix} \in R^{4\times 4}, b = \begin{bmatrix} 1 \\ 0.5 \\ 0.7 \\ 2 \end{bmatrix} \mathrm{L} = 8,\ \alpha = \frac{-7}{2},\ A = \{\frac{-7}{2}, \frac{-5}{2}, \frac{-3}{2}, \frac{-1}{2}, \frac{1}{2}, \frac{3}{2}, \frac{5}{2}, \frac{7}{2}\}$$

Problema lineal de mínimos cuadrados discreto

Algoritmo de Búsqueda exhaustiva (ML). Complejidad: O(t^n)

1. Calcular $\|Hx - b\|_2$ para todos los valores posibles de x:

   Para $i = 0, 1, 2, \ \ldots, t^n - 1$

   Calcular $x^{(i)}$

   $v(i) = \left\| Hx^{(i)} - b \right\|_2^2$

   FinPara

2. Elegir como solución el valor de x que proporciona la norma mínima

   $i = \operatorname{argmin}(v)$

   $x_{ML} = x^{(i)}$

Ejemplo:

Encontrar $x_{ML} \in A^4$ tal que $\|Hx_{ML} - b\|_2 = \min_{x \in A^4} \|Hx - b\|_2$

siendo $H = \begin{bmatrix} 1 & 0.2 & 0.3 & 0.4 \\ 0.2 & 2 & 0.5 & 0.1 \\ 0.1 & 0.9 & 3 & 0.2 \\ 0.6 & 0.7 & 0.3 & 4 \end{bmatrix} \in R^{4\times 4}, b = \begin{bmatrix} 1 \\ 0.5 \\ 0.7 \\ 2 \end{bmatrix}$ L $= 8$, $\alpha = \frac{-7}{2}$, $A = \{\frac{-7}{2}, \frac{-5}{2}, \frac{-3}{2}, \frac{-1}{2}, \frac{1}{2}, \frac{3}{2}, \frac{5}{2}, \frac{7}{2}\}$

Solución:

$x_{ML} = \begin{bmatrix} 0.5 \\ -0.5 \\ 0.5 \\ 0.5 \end{bmatrix}$

Coste aproximado:

$$(2n^2 + 2n) * t^n \ flops \approx (2 * 4^2 + 2 * 4) * 8^4 = 163840 \ flops$$

Problema lineal de mínimos cuadrados discreto

Algoritmos de resolución *Zero forcing* (subóptimo). Complejidad: $O(n^3)+O(n^2)$

1. Calcular la descomposición QR de H: $H = QR = \begin{bmatrix} R_1 \\ 0 \end{bmatrix}$. Calcular $c = Q^T b$.

$$\|Hx_{ML} - b\|_2^2 = \|QRx - b\|_2^2 = \left\|Rx - Q^T b\right\|_2^2 = \|Rx - c\|_2^2 = \|R_1 x - c_1\|_2^2 + \|-c_2\|_2^2$$

2. Resolver $R_1 x = c_1$

3. $\hat{x}_{ML} = \lfloor x \rceil$ (valor de la constelación más próximo a x)

Encontrar $x_{ML} \in A^4$ tal que $\|Hx_{ML} - b\|_2 = \min_{x \in A^4} \|Hx - b\|_2$

Ejemplo:

siendo $H = \begin{bmatrix} 1 & 0.2 & 0.3 & 0.4 \\ 0.2 & 2 & 0.5 & 0.1 \\ 0.1 & 0.9 & 3 & 0.2 \\ 0.6 & 0.7 & 0.3 & 4 \end{bmatrix} \in R^{4\times 4}, b = \begin{bmatrix} 1 \\ 0.5 \\ 0.7 \\ 2 \end{bmatrix}$ $L = 8,\ \alpha = \frac{-7}{2},\ A = \{\frac{-7}{2}, \frac{-5}{2}, \frac{-3}{2}, \frac{-1}{2}, \frac{1}{2}, \frac{3}{2}, \frac{5}{2}, \frac{7}{2}\}$

$H=QR;\ \ c=Q^Tb$

Complejidad: $(4/3)n^3+3n^2=134$ *flops*

$$R = \begin{bmatrix} -1.1874 & -0.9348 & -0.7411 & -2.3917 \\ 0 & -2.1133 & -1.5507 & -0.4847 \\ 0 & 0 & -2.5448 & 0.2177 \\ 0 & 0 & 0 & 3.1949 \end{bmatrix} \quad c = \begin{bmatrix} -1.9959 \\ -0.6455 \\ -0.3025 \\ 1.1172 \end{bmatrix} \quad x = \begin{bmatrix} 0.7923 \\ 0.1161 \\ 0.1488 \\ 0.3497 \end{bmatrix} \quad \hat{X}_{ML} = \begin{bmatrix} 0.5 \\ 0.5 \\ 0.5 \\ 0.5 \end{bmatrix}$$

Problema lineal de mínimos cuadrados discreto

Algoritmos de resolución: Sphere decoding (ML)

Idea básica: limitar la búsqueda exhaustiva a un conjunto de puntos contenidos en el interior de una esfera

Si particionamos el sistema en bloques e imponemos la condición de búsqueda dentro de la esfera se tiene:

$$\min_{s\in A^m}\|c_1-R_1 s\|_2^2=\min_{s\in A^m}\left\| \begin{bmatrix} c_k^1 \\ c_k^2 \end{bmatrix} - \begin{bmatrix} R_k^{11} & R_k^{12} \\ & R_k^{22} \end{bmatrix}\begin{bmatrix} s_k^1 \\ s_k^2 \end{bmatrix} \right\|_2^2 < r^2 \; con \; 1\leqslant k\leqslant m$$

o bien

$$\|c_1-R_1 s\|_2^2= \left\| c_k^1-(R_k^{11} s_k^1+R_k^{12} s_k^2) \right\|_2^2 + \left\| c_k^2-R_k^{22} s_k^2 \right\|_2^2 < r^2$$

Con ello obtenemos una condición más restrictiva para "podar" la rama en el caso de que las operaciones en dicho bloque superen el radio *r* estipulado por:

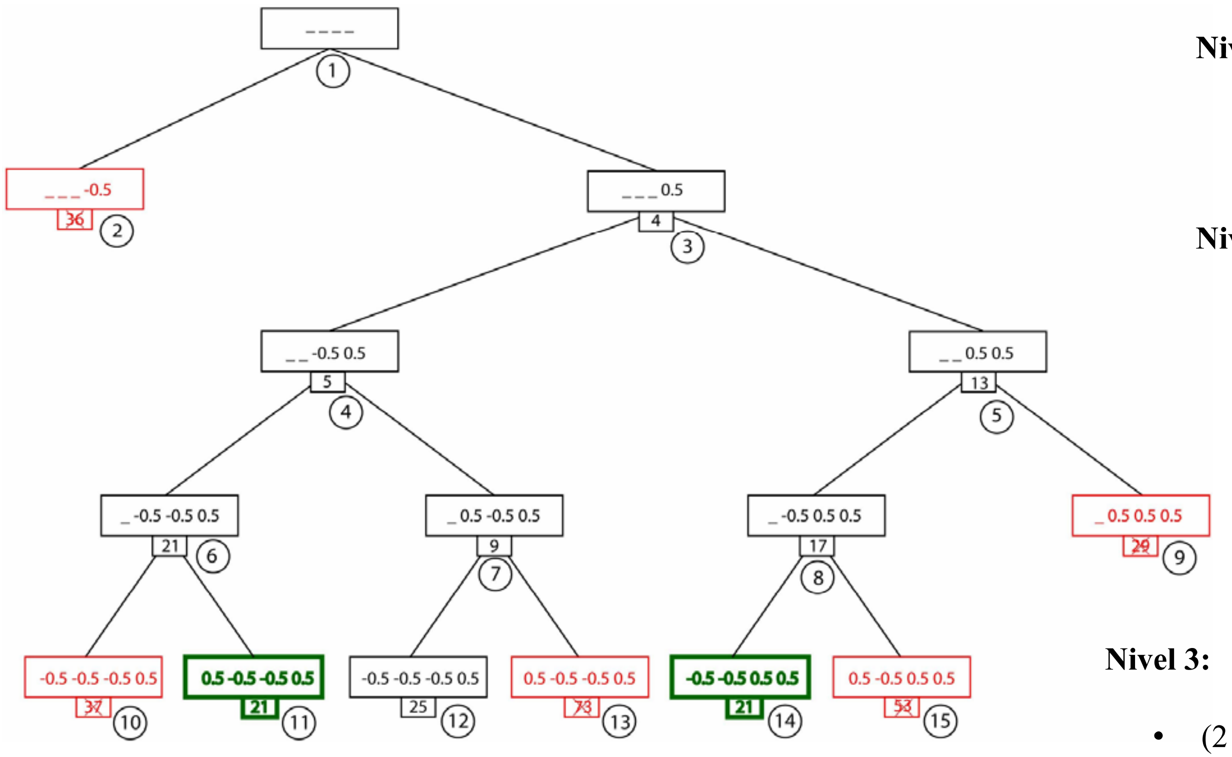
$$\left\| c_k^2-R_k^{22} s_k^2 \right\|_2^2 < r^2$$

Ejemplo

Algoritmos de resolución: Sphere decoding (ML)

$$R_1 = \begin{bmatrix} 4 & 8 & 6 & 12 \\ 0 & 6 & 2 & 4 \\ 0 & 0 & 4 & 8 \\ 0 & 0 & 0 & 4 \end{bmatrix} \qquad c_1 = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} \qquad \min_{s \in A^m} \left\| \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} - \begin{bmatrix} 4 & 8 & 6 & 12 \\ 0 & 6 & 2 & 4 \\ 0 & 0 & 4 & 8 \\ 0 & 0 & 0 & 4 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \end{bmatrix} \right\|_2$$

con una constelación de L=2 {-0.5 , 0.5} y tomando Radio=5



Nivel 1:

- $(4 - 4*(-0.5))^2 = 36 > 25$ Poda!!
- $(4 - 4*0.5)^2 = 4 < 25$

Nivel 2:

- $(3 - 4*(-0.5)-8*0.5)^2 + 4 = 5 < 25$
- $(3 - 4*0.5-8*0.5)^2 + 4 = 13 < 25$

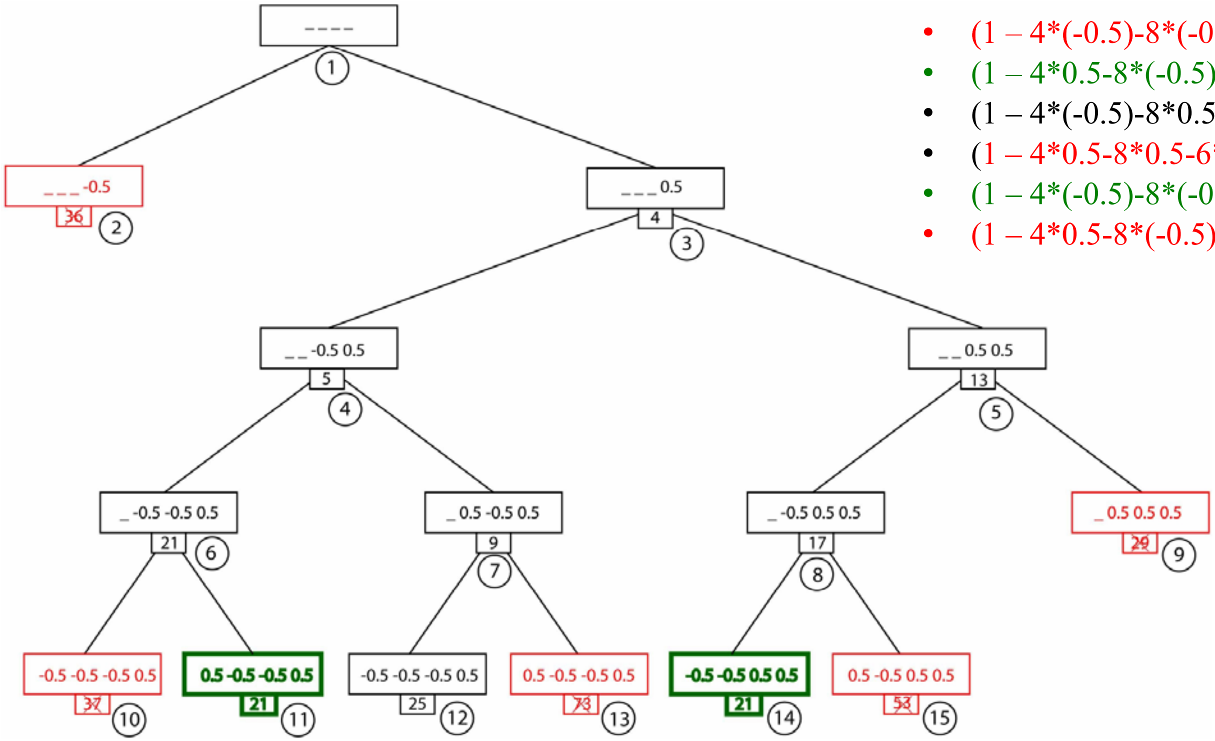
Nivel 3:

- $(2 - 6*(-0.5)-2*(-0.5)-4*0.5)^2 + 5 = 21 < 25$
- $(2 - 6*0.5-2*(-0.5)-4*0.5)^2 + 5 = 9 < 25$
- $(2 - 6*(-0.5)-2*0.5-4*0.5)^2 + 13 = 17 < 25$
- $(2 - 6*0.5-2*0.5-4*0.5)^2 + 13 = 29 > 25$ Poda!!

Ejemplo
Algoritmos de resolución: Sphere decoding (ML)

$$R_1=\begin{bmatrix}4 & 8 & 6 & 12\\ 0 & 6 & 2 & 4\\ 0 & 0 & 4 & 8\\ 0 & 0 & 0 & 4\end{bmatrix} \qquad c_1=\begin{bmatrix}1\\2\\3\\4\end{bmatrix} \qquad \min_{s\in A^m}\left\|\begin{bmatrix}1\\2\\3\\4\end{bmatrix}-\begin{bmatrix}4 & 8 & 6 & 12\\ 0 & 6 & 2 & 4\\ 0 & 0 & 4 & 8\\ 0 & 0 & 0 & 4\end{bmatrix}\begin{bmatrix}s_1\\s_2\\s_3\\s_4\end{bmatrix}\right\|_2$$

con una constelación de L=2 {-0.5 , 0.5} y tomando Radio=5



- $(1 - 4*(-0.5)-8*(-0.5)-6*(-0.5)-12*0.5)^2 + 21 = 37 > 25$ Poda!!
- $(1 - 4*0.5-8*(-0.5)-6*(-0.5)-12*0.5)^2 + 21 = 21 < 25$
- $(1 - 4*(-0.5)-8*0.5-6*(-0.5)-12*0.5)^2 + 9 = 25 = 25$
- $(1 - 4*0.5-8*0.5-6*(-0.5)-12*0.5)^2 + 9 = 73 > 25$ Poda!!
- $(1 - 4*(-0.5)-8*(-0.5)-6*0.5-12*0.5)^2 + 17 = 21 < 25$
- $(1 - 4*0.5-8*(-0.5)-6*0.5-12*0.5)^2 + 17 = 53 > 25$ Poda!!

Posibles soluciones

$$\hat{s}_1=\begin{bmatrix}0.5\\-0.5\\-0.5\\0.5\end{bmatrix} \qquad \hat{s}_2=\begin{bmatrix}-0.5\\-0.5\\0.5\\0.5\end{bmatrix}$$

Algoritmos de resolución: Sphere decoding (ML)

Elección del radio:

- Elegir un radio grande y cambiarlo de forma adaptativa al ir encontrando hojas
- Elegir como radio la distancia euclídea de una hoja elegida al azar
- Elegir como radio la distancia correspondiente a la solución *"zero forcing"*
- Otras

Preprocesado:

Calcular la descomposición QR y trabajar con una matriz triangular en lugar de hacerlo con una general:

Calcular la descomposición QR de H: $H = QR = \begin{bmatrix} R_1 \\ 0 \end{bmatrix}$. Calcular $c = Q^T b$.

$$\left\|Hx_{ML} - b\right\|_2^2 = \left\|QRx - b\right\|_2^2 = \left\|Rx - Q^T b\right\|_2^2 = \left\|Rx - c\right\|_2^2 = \left\|R_1 x - c_1\right\|_2^2 + \left\|-c_2\right\|_2^2$$

Algoritmo Sphere Decoding

Entrada: R, c, A

Salida: $x = \underset{x \in A^n}{\operatorname{argmin}} \left\| Rx - c \right\|_2^2$

1. Elegir radio r
2. Nodos=NodoInicial
3. Para nivel=1:n
4. Nodos←Expandir (Nodos, nivel)
5. Nodos ←Podar (Nodos, nivel, r)
6. FinPara
7. i=arg min(Nodos(n+1,:)
8. x=Nodos(1:n,i)

Principal dificultad: la complejidad de este algoritmo es variable

$$NodoInicial = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$nivel = 1$

Despues de Expandir: $Nodos = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -0.5 & 0.5 \\ 26 & 4 \end{bmatrix}$

Despues de Podar: $Nodos = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0.5 \\ 4 \end{bmatrix}$

Despues de FinPara: $Nodos = \begin{bmatrix} 0.5 & -0.5 & -0.5 \\ -0.5 & -05 & -0.5 \\ -0.5 & -0.5 & 0.5 \\ 0.5 & 0.5 & 0.5 \\ 21 & 25 & 21 \end{bmatrix}$

Referencias

U. Fincke, M. Pohst, *Improved methods for calculating vectors of short length in a lattice, including a complexity analysis*, Math. Comput. 44 (170) (1985) 463–471.

C.P. Schnorr, M. Euchner, *Lattice basis reduction: improved practical algorithms and solving subset sum problems*, Math. Program. 66 (1994) 181–191.

B. Hassibi, H. Vikalo, *On sphere decoding algorithm. I. Expected complexity*, IEEE Trans. Signal Process. 53 (2005) 2806–2818.

Victor M. Garcia-Molla, Antonio M. Vidal , Alberto Gonzalez, Sandra Roger. *Improved Maximum Likelihood detection through sphere decoding combined with box optimization*. Signal Processing 98 (2014) 284–294

EJERCICIOS PROPUESTOS

1. Escribe un programa en Matlab que implemente cada una de las 5 actualizaciones de la Descomposición QR.
2. Escribe un programa en Matlab que permita resolver el problema lineal de mínimos cuadrados en el caso de matrices de rango deficiente. Primero escribe un programa matlab que calcule la descomposición QR con pivotamiento mediante reflexiones de Householder.
3. Escribe un programa en Matlab que permita resolver el problema de mínimos cuadrados con restricciones de igualdad, el problema ponderado de mínimos cuadrados y el problema generalizado de mínimos cuadrados.