Universidad de Valladolid

Facultad de Ciencias

Trabajo Fin de Grado

Grado en Matemáticas

La aproximación minimax y el algoritmo de Remez. Aplicaciones.

Autor: Marta Esteban García

Tutor: Luis Mª Abia Llera

Índice general

Resumen

El proyecto hará una presentación unificada de la teoría sobre la aproximación polinómica minimax continua y discreta, y la solución minimax de sistemas lineales: condición de Haar, algoritmo del intercambio y algoritmo de Remez. Parte del proyecto es la implementación efectiva en Matlab de los distintos algoritmos para la computación de aproximaciones óptimas en la norma infinito.

El trabajo consta de tres partes:

I En el primer capítulo se repasan nociones fundamentales de teoría de la aproximación. Entre ellas se encuentran definiciones elementales como son la convexidad o envolvente convexa y ciertos teoremas de convexidad como el de Helly o el de Carathéodory, que posteriormente serán utilizados. También, ciertos resultados que garantizan la existencia y la unicidad de las mejores aproximaciones en espacios normados, algunos estudiados en el Grado.

II En el segundo capítulo se consideran los problemas de aproximación asociados a la solución de sistemas lineales de ecuaciones sobredeterminados. Cuando se trata de aproximaciones en la norma infinito, se enuncian teoremas de caracterización de la solución; se analiza el caso particular de hallar la solución minimax de un sistema de $n+1$ ecuaciones con n incógnitas. Para conseguir resolver los problemas, se estudian dos algoritmos: el ascendente y el descendente. Ambos están analizados y programados con el lenguaje de programación MATLAB.

III En el tercer y último capítulo se trata el problema general de la aproximación de una función continua en un intervalo compacto mediante un polinomio. Se habla también de un problema más general en que los polinomios serán reemplazados por otras funciones continuas. Se estudiará la teoría minimax, exponiendo el teorema de caracterización y ciertos teoremas que garantizan la unicidad y que acotan el error

en dicha teoría. Además, se explicará el algoritmo de intercambio de Remez, el cual se implementará en MATLAB.

Capítulo 1

Introducción

1.1. Conceptos básicos

Para comenzar, se definirán y enunciarán algunos resultados necesarios sobre convexidad, útiles en el desarrollo del trabajo.

Muchos problemas de teoría de la aproximación se formulan en el marco de un espacio vectorial normado. Suponemos que el lector está familiarizado ya con las propiedades fundamentales de un espacio normado y, en particular, las propiedades de la norma. Para fijar el lenguaje, en lo sucesivo $(E, \|\cdot\|_E)$ denota un espacio vectorial normado (real o complejo) E dotado de la norma $\|\cdot\|_E$.

Definición 1.1.1 (Convexidad)**.** Un conjunto $C \subset \mathbb{R}^n$ es convexo si para todo par de puntos $a, b \in C$ el segmento que los une

$$[a, b] = \{x = (1-t)a + tb \mid \forall t \in [0, 1]\}$$

está contenido en C.

De hecho, se puede verificar fácilmente que si $f_1, \ldots, f_k \in C$, y $\theta_1, \ldots, \theta_k$ son escalares no negativos con $\sum_{i=1}^{k} \theta_k = 1$, entonces $\sum_{i=1}^{k} \theta_i f_i \in C$.

Definición 1.1.2 (Envolvente convexa)**.** La envolvente convexa de un subconjunto de puntos A de un espacio vectorial es la intersección de todos los conjuntos convexos que contienen a A. Denotaremos a este conjunto por $\mathscr{H}(A)$.

Dado A en un espacio vectorial, la envolvente convexa de A viene proporcionada por el conjunto de los puntos g que se escriben como combinaciones lineales convexas de elementos de A; es decir, el conjunto de puntos

$$\mathscr{H}(A) = \{g = \sum_{i=1}^{k} \theta_i f_i : f_i \in A,\ \theta_i \in \mathbb{R},\ \theta_i \geq 0,\ \sum_{i=1}^{k} \theta_i = 1\}$$

.

El siguiente teorema expone que si el espacio vectorial es de dimensión n, cada punto g perteneciente a la envolvente convexa de A se puede expresar como una combinación lineal convexa de cierto subconjunto de a lo sumo $n+1$ puntos de A.

Teorema 1.1.3 (Teorema de Carathéodory)**.** *Sea A un subconjunto de un espacio vectorial de dimensión n. Sea x un punto cualquiera perteneciente a la envolvente convexa de A, entonces x se puede expresar como una combinación lineal convexa de $n+1$ (o menos) elementos de A.*

Demostración. Sea x un punto de $\mathscr{H}(A)$, entonces existen escalares no negativos $\theta_1, \ldots, \theta_k$ y puntos de A, $f_1, \ldots, f_k$, tales que x se puede expresar como $x = \sum_{i=1}^{k} \theta_i f_i$ y $\sum_{i=1}^{k} \theta_i = 1$. Supongamos que k es el mínimo número natural tal que x puede ser expresado de esta forma. Así todos los puntos f_i son distintos dos a dos y $\theta_i > 0$, para todo $i = 1, \ldots, k$.

Supongamos por reducción al absurdo que $k > n+1$, entonces $\{f_1, \ldots, f_k\}$ es un conjunto de más de n puntos en un espacio de dimensión n y por lo tanto, son puntos afínmente dependientes. Es decir, existen $\lambda_1, \ldots, \lambda_k \in \mathbb{R}$, no todos nulos, tales que $\lambda_1 f_1 + \ldots + \lambda_k f_k = 0$ y $\lambda_1 + \ldots + \lambda_k = 0$. Como la suma de todos los λ_i es igual a 0, y no todos ellos son nulos, habrá positivos y negativos. Además, como los θ_i son positivos, existirá $t > 0$ tal que $\theta_i + t\lambda_i \geq 0$ para todo $i = 1, \ldots, k$, con al menos uno de ellos, en que se alcanza la igualdad a cero. Así, $x = \sum_{i=1}^{k} (\theta_i + t\lambda_i) f_i$ y se consigue expresar x como una combinación lineal convexa de $k-1$ puntos de A a lo sumo, una vez eliminados los términos con coeficiente igual a 0. Sin embargo, se había supuesto que k era el mínimo natural para el que existía una combinación lineal convexa. Por lo que llegamos a una contradicción. Así pues, tendremos que $k \leq n+1$. □

Formulamos sin prueba, por ser un resultado estudiado en el grado, el teorema de existencia de un elemento de norma mínima en un conjunto convexo, cerrado y no vacío de un espacio de dimensión finita euclídeo (o un espacio de Hilbert).

Teorema 1.1.4. *Todo subconjunto cerrado convexo de un espacio Euclídeo de dimensión n posee un único punto de norma mínima.*

El siguiente teorema será fundamental para la caracterización más adelante de ciertas soluciones óptimas.

Teorema 1.1.5 (Teorema de las desigualdades lineales)**.** *Sea U un conjunto cerrado y convexo de $\mathbb{R}^n$. El sistema de desigualdades lineales $\langle u, z \rangle > 0$ $(u \in U)$ es inconsistente si y solo si $0 \in \mathscr{H}(U)$.*

Demostración. Para la suficiencia, se supone $0 \in \mathscr{H}(\mathrm{U})$. Esto implica que $0 = \sum_{i=1}^{k} \theta_i u_i$ con $\theta_i \geq 0$, $\sum_{i=1}^{k} \theta_i = 1$ y $u_i \in U$. Entonces, para todo z, $0 = \sum_{i=1}^{k} \theta_i \langle u_i, z\rangle$. Esta ecuación no se cumplirá si $\langle u_i, z\rangle > 0$ para todo $i = 1, \ldots, k$. Para la condición necesaria asumimos que $0 \notin \mathscr{H}(U)$. Existe un punto $z \in \mathscr{H}(U)$ tal que $\|z\|$ es mínimo. Sea u arbitrario en U. Por convexidad, $\theta u + (1-\theta)z \in \mathscr{H}(U)$ cuando $0 \leq \theta \leq 1$, y como consecuencia $0 \leq \|\theta u + (1-\theta)z\|^2 - \|z\|^2 = \theta^2\|u-z\|^2 + 2\theta \langle u - z, z\rangle$. Pero esta desigualdad no puede ser válida para un pequeño positivo θ, a no ser que $\langle u - z, z\rangle \geq 0$. Esto es $\langle u, z\rangle \geq \langle z, z\rangle > 0$, mostrando que z es solución de las desigualdades. □

Probaremos ahora el teorema de Helly, el cual establece condiciones para que una familia de conjuntos convexos en el espacio euclídeo tenga intersección no vacía.

Teorema 1.1.6 (Teorema de Helly)**.** *Sea $\{K_i\}_0^m$ una colección de conjuntos compactos y convexos en $\mathbb{R}^n$. Para que $\bigcap_{i=0}^m K_i$ sea no vacía, es necesario y suficiente que todas las subfamilias de $n+1$ conjuntos de $\{K_i\}$ tengan un punto en común.*

Demostración. Si $\bigcap_{i=0}^m K_i$ es no vacío, es claro que cualquier elemento en esa intersección estará en la intersección de cualquier subfamilia de $n+1$ subconjuntos de $\bigcap_{i=0}^m K_i$.

Para probar la suficiencia, sea $f(x) = \text{máx}_i\, dist(x, K_i)$. Esta función es continua por ser el máximo de un número finito de funciones continuas; las definidas por $x \to d(x, K_i)$, $i = 0, \ldots, m$. Como cada K_i es compacto, y por tanto acotado, se tiene que $d(x, K_i)$ tiende a infinito cuando $\|x\| \to \infty$.

Razonemos por reducción al absurdo y supongamos que $\bigcap_{i=0}^m K_i$ es vacío. Puesto que $f(x) \to \infty$ si $\|x\| \to \infty$, el ínfimo de f se alcanzará en algún punto y, y por ser $\bigcap K_i = \varnothing$, necesariamente debemos tener $\rho = f(y) > 0$. Se puede suponer, sin pérdida de generalidad, que $y = 0$, ya que, si no se cumpliera, bastaría una traslación para trasladar el origen de coordenadas al punto y. Tomemos para cada i un $x_i \in K_i$ tal que $\|x_i\| = dist(y, K_i) = dist(0, K_i)$ y, por tanto, tal que $\rho = \text{máx}_{1 \leq i \leq m} \|x_i\|$. La igualdad se alcanza en varios de esos puntos, y sin pérdida de generalidad puedo suponer que la igualdad ocurre en los primeros puntos $x_0, \ldots, x_q$, mientras que en el resto de los puntos se tiene $\|x_i\| < \rho$, $i = q+1, \ldots, m$. Si $0 \notin \mathscr{H}\{x_0, ..., x_q\}$, entonces $f(y)$ puede decrecer moviéndose ligeramente hacia $\mathscr{H}\{x_0, ..., x_q\}$. En consecuencia, $0 \in \mathscr{H}\{x_0, ..., x_q\}$. Por el teorema de Carathéodory, se puede escribir $0 = \sum_{i=0}^{n+1} \lambda_i x_i$ con $\sum_{i=0}^{n+1} \lambda_i = 1,\ \lambda_i \geq 0$.

Sea z arbitrario, entonces

$$0 = \langle 0, z\rangle = \left\langle \sum_{i=0}^{n+1} \lambda_i x_i, z \right\rangle = \sum_{i=0}^{n+1} \lambda_i \langle x_i, z\rangle \geq \sum_{i=0}^{n+1} \lambda_i \|x_i\| \rho$$

que contradice que $\rho > 0$ y $\lambda_i, \|x_i\| > 0$ $\forall i$. Se obtiene por tanto que el sistema $\langle z, x_i\rangle \geq 0$ $(i = 0, ..., q)$ es inconsistente. Pero cada punto de K_i, $(i = 1, ..., q)$ satisface $\langle z, x_i\rangle \geq \rho^2$. Por tanto, $\bigcap_{i=0}^{q} K_i = \varnothing$. □

1.2. Existencia y unicidad de aproximaciones óptimas

El problema general que se propone es determinar un punto o puntos de un subconjunto dado S de un espacio métrico, que esté a una distancia mínima de un punto fijo dado. En general, estos puntos más próximos podrían no existir. Únicamente añadiendo hipótesis adicionales sobre el subconjunto S o el espacio métrico, se puede establecer un teorema de existencia. En general la existencia se derivará de alguna condición de compacidad del conjunto de aproximantes, mientras que la unicidad estará asociada a propiedades de convexidad del espacio. La atención en esta sección se centrará en los espacios normados. El siguiente teorema establece la compacidad de los cerrados y acotados que se incluyen en subespacios de dimensión finita de un espacio normado.

Teorema 1.2.1. *Todo subconjunto cerrado, acotado de un subespacio de dimensión finita en un espacio vectorial normado es compacto.*

Demostración. Sea F un conjunto cerrado, acotado y de dimensión finita contenido en un espacio normado E. Entonces existe un conjunto linealmente independiente $\{g_1, \ldots, g_n\}$ con la propiedad de que cada elemento $f \in F$ se puede expresar de forma única de la siguiente manera: $f = \sum \lambda_i g_i$. Sea T la aplicación lineal de $\mathbb{R}^n \to E$ que aplica $\lambda = (\lambda_1, \ldots, \lambda_n) \to f$. Dotamos a $\mathbb{R}^n$ con la norma $\|\lambda\| = \max |\lambda_i|$, entonces T es continua, debido a

$$\begin{aligned} \|T\lambda\|_E &= \|\sum_{i=1}^{n} \lambda_i g_i\| \leq \sum_{i=1}^{n} \|\lambda_i\, g_i\| \leq \sum_{i=1}^{n} |\lambda_i|\, \|g_i\| \\ &\leq (\max_i |\lambda_i|) \sum_{i=1}^{n} \|g_i\| = \|\lambda\|_\infty \sum_{i=1}^{n} \|g_i\|. \end{aligned}$$

El conjunto F es la imagen bajo la aplicación T, del conjunto

$$M = T^{-1}(F) = \{\lambda = (\lambda_1, \ldots, \lambda_n) : T\lambda \in F\}$$

Si se demostrara que M es compacto, se obtendría que F también lo es. Para ver que M es compacto, basta probar que es un conjunto cerrado y acotado de $\mathbb{R}^n$, en virtud del teorema de Bolzano-Weierstrass. Si $\{\lambda^{(k)}\}$ es una sucesión de elementos de M y $\lambda^{(k)} \to \lambda$ cuando $k \to \infty$, entonces $T\lambda \in F$ porque $T\lambda = \lim_{k\to\infty} T\lambda_k \in F$, y F es cerrado, por tanto, $\lambda \in M$. Esto muestra que M es cerrado. Falta probar que es acotado. Como el conjunto $\{\lambda : \|\lambda\| = 1\} \in \mathbb{R}^n$ es compacto y T es continuo, el ínfimo, α, de $\|T\lambda\|$ se alcanza en ese conjunto. Debido a que $\{g_1, \ldots, g_n\}$ es linealmente independiente, $\alpha > 0$. Por ello, para cualquier $\lambda \neq 0$, $\|T\lambda\| = \|T\left(\frac{\lambda}{\|\lambda\|}\right)\| \cdot \|\lambda\| \geq \alpha\|\lambda\|$. Por tanto, $\|T\lambda\|$ está acotado en M y $\|\lambda\|$ está acotada en M. □

Enunciamos primero un teorema de existencia para espacios métricos generales.

Teorema 1.2.2 (Teorema de existencia de la mejor aproximación en un espacio métrico)**.** *Sea K un conjunto compacto en un espacio métrico E. Para cada punto p perteneciente al espacio E existe al menos un punto $x \in K$ a distancia mínima de p; es decir, tal que $d(p, x) = d(p, K) = \inf_{y\in K} d(p, y)$.*

Demostración. El resultado es consecuencia inmediata del teorema de Weierstrass, por el que el inferior de una función continua definida en un compacto siempre se alcanza en el compacto. Aquí la función que se considera es $f(y) = d(p, y)$, con $y \in K$. □

En el caso particular en que S es un subespacio vectorial de dimensión finita de un espacio normado, se tiene:

Teorema 1.2.3 (Teorema de la existencia)**.** *Sea M un subespacio vectorial con dimensión finita de un espacio vectorial normado E. Entonces, fijado un punto $g \in E$, se puede encontrar, al menos, un punto $f \in M$ a distancia mínima de g.*

Demostración. Sea M tal subespacio y sea g el punto dado. Sea f_0 un punto arbitrario de M. Entonces el punto buscado pertenece al conjunto:

$$K = \{f : f \in M,\ \|f - g\| \leq \|f_0 - g\|\}.$$

Este conjunto es cerrado y acotado en M, de dimensión finita, y por el teorema 1.2.1, debe ser compacto. Entonces el teorema 1.2.2 da la existencia de un punto de K a distancia mínima de g, como se quería probar. □

La hipótesis de la finitud del espacio es necesaria y se verá con un contraejemplo. El ejemplo servirá para mostrar que la hipótesis de dimension finita no puede ser omitida en el teorema anterior.

Ejemplo 1.2.4. Sea (c_0) el espacio de sucesiones infinitas $f = (\xi_1, \xi_2, \ldots)$ tales que $\xi_n \to 0$ y con la norma $\|f\| = \text{máx}\,|\xi_n|$. Se considera el subespacio M contenido en (c_0), de puntos f para los cuales $\sum_{k=1}^{\infty} 2^{-k}\xi_k = 0$. Probar que este subespacio M no contiene un punto más cercano a ningún punto $g \in (c_0)$ externo de M.

Sea $g = (\eta_1, \eta_2, \ldots)$ cualquier punto de (c_0) que no se encuentra en M. Entonces, el número $\lambda = \sum_{k=1}^{\infty} 2^{-k}\eta_k$ es distinto de cero. Se prueba primero que la distancia desde g a M no es mayor que $|\lambda|$. De hecho, los siguientes puntos pertenecen a M:

$$\begin{aligned} f_1 &= -\frac{2}{1}(\lambda, 0, 0, \ldots) + g \\ f_2 &= -\frac{4}{3}(\lambda, \lambda, 0, 0, \ldots) + g \\ f_3 &= -\frac{8}{7}(\lambda, \lambda, \lambda, 0, 0, \ldots) + g \\ &\vdots \end{aligned}$$

Si se continuase, se obtendría: $\|f_n - g\| = \frac{2^n}{2^n - 1}|\lambda| \to |\lambda|$. Faltaría por probar que no hay ningún punto de M con distancia $|\lambda|$ o menor desde g. Si se toma $f = (\xi_1, \xi_2, \ldots)$ arbitrario en M, se puede seleccionar n tal que $|\xi_k - \eta_k| < \frac{1}{2}|\lambda|$ con $k \geq n$. Esto es posible debido a que los elementos de (c_0) son sucesiones que convergen a cero. Suponiendo $\|g - f\| \leq |\lambda|$, entonces

$$\begin{aligned} \left|\sum_{k=1}^{\infty} 2^{-k}\eta_k\right| &= \left|\sum_{k=1}^{\infty} 2^{-k}(\eta_k - \xi_k)\right| \leq \sum_{k=1}^{\infty} 2^{-k}|\eta_k - \xi_k| \leq \\ &\leq |\lambda| \sum_{k<n} 2^{-k} + \frac{1}{2}|\lambda| \sum_{k \geq n} 2^{-k} \ < |\lambda| \end{aligned}$$

Lo que nos lleva a una contradicción.

Damos ahora un teorema de existencia y unicidad de mejores aproximaciones que requiere de varias propiedades de convexidad. Empezamos enunciando la definición de espacio uniformemente convexo.

Definición 1.2.5. Un espacio vectorial normado E es uniformemente convexo si para cada $\epsilon > 0$ le corresponde un $\delta > 0$ tal que $\|f - g\| < \epsilon$ cuando $\|f\| = \|g\| = 1$ y $\|\frac{1}{2}(f + g)\| > 1 - \delta$.

Geométricamente, la definición dice que para todo segmento con extremos en la esfera unidad si su punto medio tiene magnitud suficientemente cerca de la unidad entonces los extremos del segmento están arbitrariamente próximos.

Teorema 1.2.6. *Sea K conjunto cerrado y convexo en un espacio uniformemente convexo de Banach E. Fijado un punto g, el conjunto K posee un único punto cercano al punto dado.*

Demostración. Sea K un conjunto cerrado y convexo en un espacio uniformemente convexo de Banach E y sea g un punto arbitrario. Haciendo el cambio de variable: $f \to f - g$ se puede asumir, sin pérdida de generalidad, que $g = 0$. Denotemos con $D = \text{ínf}_{f\in K} \|f-g\|$. Si $D = 0$, como K es cerrado, $g \in K$ y, por tanto, la conclusión es obvia. En caso de que $D \neq 0$, haciendo el cambio de variable $f \to D^{-1}f$ se puede asumir $D = 1$. Seleccionemos entonces $f_n \in K$ tal que $\lim_{n\to\infty} \|f_n\| = 1$. Sea $\epsilon > 0$, y tomemos δ como en la definición 1.2.5. Se selecciona N para que se cumpla $\|f_n\| - 1 < \delta$ siempre que $n \geq N$. Sean $n, m \geq N$, y para simplificar la notación, sea $\lambda_n = \|f_n\|^{-1}$. Entonces, usando la desigualdad triangular y la convexidad de K, se obtiene:

$$\begin{aligned}\frac{1}{2}\|\lambda_n f_n + \lambda_m f_m\| &= \frac{1}{2}\|f_n + f_m - (1-\lambda_n)f_n - (1-\lambda_m)f_m\| \\ &\geq \frac{1}{2}\|f_n + f_m\| - \frac{1}{2}(1-\lambda_n)\|f_n\| - \frac{1}{2}(1-\lambda_m)\|f_m\| \\ &> 1 - \delta,\end{aligned}$$

donde hemos utilizado en la última desigualdad que $1/2\|f_n + f_m\| \geq 1$ por ser un elemento de K.

Por la convexidad uniforme, $\|\lambda_n f_n - \lambda_m f_m\| < \epsilon$, y por tanto $\{\lambda_n f_n\}$ es una sucesión de Cauchy. Por ser E un espacio completo, $\lambda_n f_n \to f$ para alguna f. Como $\|f_n - f\| \leq \|f_n - \lambda_n f_n\| + \|\lambda_n f_n - f\| \leq \|f_n\|(1-\lambda_n) + \|\lambda_n f_n - f\|$, se tiene que $f_n \to f$, ya que $\|f_n\| - 1$ y $\lambda_n f_n - f\|$ pueden hacerse arbitrariamente pequeños sin más que tomar n suficientemente grande. Por ser K cerrado, $f \in K$, y como $\|\lambda_n f_n\| = 1$, se tiene también que $\|f\| = 1$, es decir, $f \in K$ está a distancia mínima de g.

Para ver la unicidad, se usa la reducción al absurdo. Sean dos puntos f, h ambos pertenecientes a K y que verifican $\|g - f\| = \|g - h\| = \lambda$, siendo λ la distancia mínima. Usando la ley del paralelogramo:

$$\|f - h\|^2 + 4\|g - \left(\frac{f+h}{2}\right)\|^2 = 2\|f - g\|^2 + 2\|h - g\|^2 = 4\lambda^2$$

Como $\frac{f+h}{2} \in K$ se obtiene:

$$\|g - \left(\frac{f+h}{2}\right)\| \geq \lambda$$

que contradice el que E es un espacio uniformemente convexo. □

El teorema que sigue nos garantiza la unicidad de la mejor aproximación en subespacios cuando el espacio normado es estrictamente convexo.

Teorema 1.2.7 (Teorema de la unicidad)**.** *Dado un punto cualquiera g en un espacio vectorial normado estrictamente convexo, existe un único punto f perteneciente a un subespacio M de dimensión finita y tal que f es el más cercano a g.*

Demostración. La existencia ya ha sido probada en el teorema 1.2.3. Para ver la unicidad razonaremos mediante el método del absurdo. Sean f y f' dos puntos del subespacio M con mínima distancia λ al punto inicial g. Entonces,

$$\|\frac{1}{2}(f+f') - g\| \leq \frac{1}{2}\|f-g\| + \frac{1}{2}\|f'-g\| = \lambda$$

Por ser M un subespacio vectorial, $\frac{1}{2}(f+f') \in M$, y en consecuencia, se obtiene que $\frac{1}{2}\|f-g\| + \frac{1}{2}\|f'-g\| \geq \lambda$. Si $\lambda = 0$, es obvio que $f = f' = g$. Si $\lambda \neq 0$, entonces los vectores $(f-g)/\lambda$, $(f'-g)/\lambda$ y sus puntos medios son todos de norma 1, y por ser estrictamente convexo, se obtiene, $f = f'$. □

Teorema 1.2.8. *Sea K un conjunto compacto en un espacio métrico X. Si para cada punto x en X hay un único punto que es el más cercano $\mathfrak{J}x$ en K, entonces $\mathfrak{J}$ es un operador continuo.*

Demostración. Si $\mathfrak{J}$ es discontinuo en x_0, entonces existe $\epsilon > 0$ y una sucesión $x_n \to x_0$ tal que $d(\mathfrak{J}x_n, \mathfrak{J}x_0) > \epsilon$. Pasando a una subsucesión, si es necesario, podemos asumir que $\mathfrak{J}x_n$ converge a un punto $z \neq \mathfrak{J}x_0$. Ahora, $d(x, \mathfrak{J}x)$ es una función continua de x, como se puede ver en la siguiente desigualdad:

$$d(x, \mathfrak{J}x) \leq d(x, \mathfrak{J}y) \leq d(x,y) + d(y, \mathfrak{J}y)$$

En consecuencia, si se pasa al límite en la desigualdad,

$$d(x_0, z) \leq d(x_0, x_n) + d(x_n, \mathfrak{J}x_n) + d(\mathfrak{J}x_n, z)$$

se obtiene $d(x_0, z) \leq d(x_0, \mathfrak{J}x_0)$, lo cual contradice la unicidad de $\mathfrak{J}x_0$. □

Capítulo 2

Solución minimax de sistemas de ecuaciones lineales

En este capítulo se considerarán los problemas de aproximación minimax que surgen de los sistemas de ecuaciones lineales sobredeterminados:

$$\sum_{j=1}^{n} A_j^i x_j = b_i \quad (i = 1, \ldots, m) \tag{2.1}$$

Los datos A_j^i y b_i vienen dados por el problema y las incógnitas son las x_j, $j = 1, \ldots, n$. Un sistema como el (2.1) se puede reinterpretar de la siguiente forma: si se ve el primer miembro de (2.1) como una combinación lineal de las columnas $\mathbf{A}_j = [A_j^1, A_j^2, \ldots, A_j^m]^T$, $j = 1, \ldots, n$, de la matriz

$$A = \begin{bmatrix} A_1^1 & A_2^1 & \cdots & A_n^1 \\ A_1^2 & A_2^2 & \cdots & A_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ A_1^m & A_2^m & \cdots & A_n^m \end{bmatrix}$$

el problema de aproximación que se plantea es determinar los coeficientes $\mathbf{x} = [x_1, \ldots, x_n]^T$ tal que la norma

$$\|\mathbf{b} - \sum_{j=1}^{n} \mathbf{A}_j x_j\|_\infty$$

sea mínima, siendo $\mathbf{b} = [b_1, \ldots, b_m]^T$. Este sistema de ecuaciones puede no tener solución (sistemas incompatibles), puede tener exactamente una solución (sistemas compatibles determinados) o puede tener varias soluciones (sistemas compatibles indeterminados). Introducimos también una notación específica para los vectores fila de la matriz A: $\mathbf{A}^i = [A_1^i, \ldots, A_n^i]$, $i = 1, \ldots, n$.

Este capítulo versa sobre el cálculo de una solución aproximada a (2.1) cuando el sistema es incompatible. Para resolver nuestro problema se debe intentar que el vector $\sum_{j=1}^{n} \mathbf{A}_j x_j - \mathbf{b}$ sea lo más próximo a cero, o en otros términos, se intenta encontrar el elemento del subespacio de $\mathbb{R}^m$ generado por las columnas de A que está a mínima distancia en la norma infinito del vector $\mathbf{b}$.

Ya vimos en el teorema de existencia 1.2.3 que para cualquier norma definida en $\mathbb{R}^m$ existe solución para este problema. Distintas normas dan origen a distintos problemas de aproximación. En este capítulo se hablará sobre el problema asociado con la norma:

$$\|\mathbf{y}\|_\infty = \max_{1\leq i\leq m} |y_i|$$

A esta norma se la llama del máximo, norma uniforme o norma Chebyshev, en honor del matemático que inició el estudio de este problema de aproximación en torno a 1850. Cuando se utiliza la norma infinito, la solución mejor aproximación de (2.1) es cualquiera que minimice la expresión:

$$\Delta(x) = \max_{1\leq i\leq m} |\sum_{j=1}^{n} A_j^i x_j - b_i| \tag{2.2}$$

Debido a ello, $\mathbf{x}$ es llamada una *solución minimax* del sistema. También es posible considerar otro problema similar y, en vez de minimizar la expresión (2.2), se puede minimizar la siguiente:

$$\delta(x) = \max_i \left\{ \sum_{j=1}^{n} A_j^i x_j - b_i \right\} \tag{2.3}$$

El problema de Tchebycheff está incluido en esta nueva formulación (en la que no aparecen los valores absolutos) de la siguiente manera. Se duplica el número de datos poniendo:

$$A^{i+m} = -A^i \text{ y } b_{i+m} = -b_i$$

Si además se pone $r_i = \sum_{j=1}^{n} A_j^i x_j - b_i$, entonces $r_{i+m} = -r_i$ y $\max_{1\leq i\leq m} |r_i| = \max_{1\leq i\leq 2m} r_i$.

2.1. Caracterización de la solución

El objetivo de este apartado es desarrollar métodos para localizar los puntos mínimos de la función $\delta(x)$ introducida al final de la sección anterior. Se va a enunciar una serie de propiedades que distinguen a la solución de cualquier otro punto.

Teorema 2.1.1 (Teorema de caracterización)**.** *Un punto z es un punto mínimo de la función δ si y solo si el origen de coordenadas en $\mathbb{R}^n$ pertenece a la envolvente convexa del conjunto de vectores fila $\{\mathbf{A}^i : r_i(z) = \delta(z)\}$.*

Demostración. Supongamos que z no es un punto mínimo de δ. Entonces, existe algún vector $\mathbf{h}$ tal que $\delta(z-h) < \delta(z)$. Sea $M = \{i : \ r_i(z) = \delta(z)\}$. Entonces, para $i \in M$, se tiene $r_i(z-h) \leq \delta(z-h) < \delta(z) = r_i(z)$ y, por tanto, $\left\langle \mathbf{A^i}, z-h \right\rangle - b_i < \left\langle \mathbf{A^i}, z \right\rangle - b_i$. Por lo que, resolviendo la inecuación anterior, se obtiene que $\left\langle \mathbf{A^i}, h \right\rangle > 0$ $(i \in M)$. Por el teorema de las desigualdades lineales 1.1.5, el origen no pertenece a la envolvente convexa del conjunto $\left\{\mathbf{A^i} : \ i \in M\right\}$.

Para ver la otra implicación, suponiendo que 0 no pertenece a la envolvente convexa $\left\{\mathbf{A^i} : \ i \in M\right\}$, se obtiene de nuevo por el teorema de las desigualdades lineales que existe h tal que $\left\langle \mathbf{A^i}, h \right\rangle > 0$ para $i \in M$ por lo que existe un número α tal que $\alpha = \text{mín}_{i \in M} \left\langle \mathbf{A^i}, h \right\rangle$ con α positivo. Los residuos $r_i(z)$, para $i \in M$, decrecen en la dirección $-h$ debido a que para todo λ positivo,

$$r_i(z - \lambda h) = r_i(z) - \lambda \left\langle \mathbf{A^i}, h \right\rangle \leq \delta(z) - \lambda\alpha$$

luego z no es mínimo. Los residuos $r_i(z)$, para $i \notin M$, son menores estrictamente que $\delta(z)$, y por la continuidad permanecen así en un entorno de z. Por lo tanto, hay puntos cercanos a z cuyos residuos toman valores menores que δ. Sea $\beta = \text{máx}_{i \in M} r_i(z)$ y $\gamma = \text{mín}_{1 \leq i \leq m} \langle \mathbf{A}^i, h \rangle$. Entonces para $i \notin M$,

$$r_i(z - \lambda h) = r_i(z) - \lambda \left\langle \mathbf{A}^i, h \right\rangle \leq \beta - \lambda\gamma$$

Se pueden hacer los residuos menores que $c = \frac{1}{2}\left[\beta + \delta(z)\right]$, seleccionando $\lambda > 0$ de tal manera que

$$\beta - \lambda\gamma < c \ \text{ y } \ \delta(z) - \lambda\alpha < c$$

con lo que z no puede ser mínimo de δ. □

Para la función Δ se obtiene un teorema similar con demostración análoga en el que se debe usar la función sgn, definida como:

$$sgn(x) = \begin{cases} -1 & \text{si } x < 0, \\ 0 & \text{si } x = 0, \\ 1 & \text{si } x > 0. \end{cases}$$

Teorema 2.1.2 (Teorema de caracterización)**.** *Dado un punto $z \in \mathbb{R}^n$, sea $\sigma_i = sgn\ r_i(z)$ y $M = \{i : \ |r_i(z)| = \Delta(z)\}$. El punto z minimiza Δ si y solo si, el origen de $\mathbb{R}^n$ pertenece a la envolvente convexa del conjunto $\{\sigma_i \mathbf{A}^i : i \in M\}$.*

Los siguientes teoremas serán esenciales en la búsqueda de la solución minimax.

Teorema 2.1.3. *Si x es un punto mínimo de la función*

$$\delta(x) = \max_{1 \le i \le m} r_i(x),$$

entonces z es un punto mínimo de $\max_{i \in J} r_i(x)$ *donde J es un subconjunto de $\{1, \ldots, m\}$ que consta de al menos $n+1$ índices.*

Demostración. Por el teorema de caracterización 2.1.1 sabemos que el origen de $\mathbb{R}^n$ permanece en la envolvente convexa del conjunto $\{\mathbf{A}^i : i \in M\}$, donde $M = \{i : r_i(z) = \delta(z)\}$. Si M contiene $n+1$ o menos elementos, consideramos $J = M$. En el caso en que M tenga más de $n+1$ elementos, utilizando el teorema de Caratheodory, seleccionamos un subconjunto J de M que tenga un máximo de $n+1$ elementos y tal que $0 \in \mathscr{H}\{\mathbf{A}^i : i \in J\}$. Por el teorema de caracterización, z es también un punto mínimo de $\max_{i \in J} r_i(x)$. □

Tendríamos también el siguiente resultado análogo para la función Δ.

Teorema 2.1.4. *Cada solución minimax del sistema* $\sum_{j=1}^{n} A_j^i x_j = b_i \ \ (i = 1, \ldots, m > n)$ *es una solución minimax de un cierto subsistema que tiene n+1 ecuaciones.*

Una vez que se encuentra tal subsistema apropiado, es sencillo encontrar su solución minimax. En diversos métodos, el encontrar este subsistema es lo más complejo.

2.2. El caso especial, m = n + 1

En esta sección se hablará del problema de hallar la solución minimax de un sistema de $n+1$ ecuaciones con n incógnitas.

$$r_i(x) = \sum_{j=1}^{n} A_j^i x_j - b_i = \left\langle \mathbf{A}^i, x \right\rangle - b_i = 0 \quad (i = 1, \ldots, n+1)$$

Hay distintos métodos para resolver tal sistema. El primero que vamos a exponer es el método de La Vallée Poussin.

2.2.1. El método de La Vallée Poussin

Supongamos que se es capaz de descubrir un punto x, signos $\sigma_i = \pm 1$, y un número ϵ tal que:

$$r_i(x) = \sigma_i \epsilon \quad (i = 1, \ldots, n+1) \tag{2.4}$$

y

$$0 \in \mathscr{H}\left\{\sigma_1 \mathbf{A}^1, \ldots, \sigma_{n+1}\mathbf{A}^{n+1}\right\} \tag{2.5}$$

Afirmamos entonces que x es una solución minimax del sistema. De hecho, por (2.4) se obtiene que $|r_i(x)| = |\epsilon|$. Y, por el teorema de caracterización 2.1.2 y por la propiedad (2.5), x es la solución buscada.

Primero, se debe encontrar una solución no trivial del sistema de ecuaciones lineales $\sum_{i=1}^{n+1} \lambda_i \mathbf{A}^i = 0$. Esto es posible debido a que los $n+1$ vectores $\mathbf{A}^1, \ldots, \mathbf{A}^{n+1}$ son necesariamente linealmente dependientes por ser elementos del espacio de dimensión n. Si se define $\sigma_i = 1$ cuando $\lambda_i \geq 0$ y $\sigma_i = -1$ cuando $\lambda_i < 0$, la condición (2.5) se cumple debido a $0 = \sum(\sigma_i \lambda_i)(\sigma_i \mathbf{A}^i)$.

Supongamos ahora que la matriz es de rango n. Aunque algún subconjunto de n filas pueda ser linealmente dependiente, se pueden renumerar las filas y tomar el conjunto $\{\mathbf{A}^1, \ldots, \mathbf{A}^n\}$ como el formado por n filas independientes. Si se cumple la condición (2.4), entonces

$$\left\langle \mathbf{A}^i, x \right\rangle - b_i = \epsilon \sigma_i.$$

Multiplicando esta ecuación por λ_i y sumando $i = 1, \ldots, n+1$

$$\left\langle \sum_{i=1}^{n+1} \lambda_i \mathbf{A}^i, x \right\rangle - \sum_{i=1}^{n+1} \lambda_i b_i = \epsilon \sum_{i=1}^{n+1} \sigma_i \lambda_i.$$

A la vista de lo que se sabe, la ecuación anterior se reduce a $-\sum \lambda_i b_i = \epsilon \sum |\lambda_i|$, y esta ecuación se puede tomar como la definición de ϵ ya que $\sum |\lambda_i| > 0$.

Faltaría por ver que con esta definición de ϵ, la ecuación (2.4) es consistente. Si no se usa la ecuación que corresponde a $i = n+1$, el sistema restante se puede resolver para una única x, debido a la suposición de que el conjunto $\{\mathbf{A}^1, \ldots, \mathbf{A}^n\}$ es linealmente independiente. Por tanto, se tiene que $r_i(x) = \sigma_i \epsilon$ para $i = 1, \ldots, n$. Como ya se ha visto,

$$\sum_{i=1}^{n+1} \lambda_i r_i(x) = \epsilon \sum_{i=1}^{n+1} \lambda_i \sigma_i(x).$$

Por tanto, $\lambda_{n+1} r_{n+1}(x) = \epsilon \sigma_{n+1} \lambda_{n+1}$. Si $\lambda_{n+1} = 0$, entonces la ecuación $\sum \lambda_i \mathbf{A}^i = 0$ representaría una dependencia lineal entre $\mathbf{A}_1, \ldots, \mathbf{A}_n$, lo cual está descartado por hipótesis. Por tanto, $\lambda_{n+1} \neq 0$ y $r_{n+1}(x) = \epsilon \sigma_{n+1}$.

Una observación importante es que para un sistema de ecuaciones $(n+1)\times n$ de rango n, una vez que los signos σ_i son conocidos, la solución minimax se puede obtener resolviendo un conjunto consistente de $n+1$ ecuaciones lineales de $n+1$ incógnitas. En problemas de este tipo provenientes de la aproximación de funciones continuas en un intervalo, los signos pueden determinarse fácilmente sin resolver ninguna ecuación lineal. En el caso general del que se habla aquí, sin embargo, es necesario determinar los signos σ_i de manera diferenciada de las x_i o de ϵ.

2.2.2. Método usando los cofactores

Este segundo método es en teoría el mismo que el descrito previamente, solo difiere en que se intenta realizar menos operaciones. Seguimos considerando A como la matriz dada $(n+1)\times n$ de rango n y $\mathbf{b}$ será la matriz columna de $n+1$ elementos. Recordemos que lo que buscamos son los $x_1, x_2, \ldots, x_n$, tales que

$$\max_i |\sum_{j=1}^{n} A_j^i x_j - b_i| = \delta \tag{2.6}$$

donde δ es el valor mínimo entre todas las posibles elecciones de $x_1, \ldots, x_n$.

Denotaremos por $\mathscr{U}$ a la matriz cuadrada formada añadiendo la $(n+1)$-ésima columna a A.

$$\mathscr{U} = \begin{bmatrix} A_1^1 & A_2^1 & \cdots & A_n^1 & A_{n+1}^1 \\ A_1^2 & A_2^2 & \cdots & A_n^2 & A_{n+1}^2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ A_1^m & A_2^m & \cdots & A_n^m & A_{n+1}^m \end{bmatrix}$$

con $A_{n+1}^i = sgn\ \mathscr{U}_{n+1}^i$, $(i = 1, 2, \ldots, n+1)$ donde $\mathscr{U}_{n+1}^i$ es el cofactor de A_{n+1}^i y sgn es su signo. Para empezar, suponemos que las primeras n filas de A son linealmente independientes. Se debe tener en cuenta que inicialmente, no se conocen los valores de $A_{n+1}^1, A_{n+1}^2, \ldots, A_{n+1}^{n+1}$ y calcularlos mediante su definición requeriría de un cálculo costoso.

Usando un procedimiento similar a una eliminación Gaussiana, la matriz $\mathscr{U}$ puede ser transformada mediante operaciones elementales en las columnas (o bien intercambiándolas o bien sumando un múltiplo de una columna a

otra) hasta adquirir la siguiente forma:

$$\begin{bmatrix} d_1 & 0 & \cdots & 0 & A^1_{n+1} \\ 0 & d_2 & \cdots & 0 & A^2_{n+1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & d_n & A^n_{n+1} \\ e_1 & e_2 & \cdots & e_n & A^{n+1}_{n+1} \end{bmatrix} = \mathscr{U} T_1 \tag{2.7}$$

Aquí, se cumple que $d_1, d_2, \ldots, d_n$ son no nulos y T_1 es el producto de transformaciones elementales de Gauss en las columnas.

A lo largo de todo el proceso, los elementos de la $(n+1)$-ésima columna de $\mathscr{U}$ permanecen sin determinar pero invariables. Una operación elemental en la que se suma un múltiplo de una columna a otra, no modifica ninguno de los cofactores $\mathscr{U}^i_{n+1}$, mientras que si se intercambia alguna columna, el signo de cada uno de los cofactores cambia. Por lo tanto, resolver el problema minimax implica conocer los A^i_{n+1} en términos de la nueva matriz $\mathscr{U} T_1$ calculada. Si denotamos D_i el cofactor de A^i_{n+1} en $\mathscr{U} T_1$, entonces, $D_{n+1} = d_1 d_2 \cdots d_n$.

$$D_i = (-1)^{n+i+1} \begin{vmatrix} d_1 & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 \\ 0 & d_2 & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \ddots & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & 0 & d_{i-1} & 0 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & 0 & d_{i+1} & 0 & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & 0 & d_{i+2} & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \ddots & \cdot & 0 \\ 0 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 & d_n \\ e_1 & e_2 & \cdots & e_{i-1} & e_i & e_{i+1} & \cdots & e_{n-1} & e_n \end{vmatrix} \quad i = 1, 2, \ldots, n.$$

La expresión para D_i puede escribirse usando los menores de la i-ésima columna, lo que resulta:

$$D_i = (-1)^{n+i+1}(-1)^{n+i} e_i d_1 d_2 \cdots d_{i-1} d_{i+1} \cdots d_n.$$

Debido a que $d_i \neq 0$ podemos resumir los resultados, de forma que:

$$\begin{array}{rcll} D_{n+1} & = & d_1 d_2 \cdots d_n & \\ D_i & = & -D_{n+1} e_i / d_i & (i = 1, \ldots, n). \end{array}$$

Usando ahora la regla de Cramer y conociendo los valores de D_i, podremos resolver δ:

$$\delta = \frac{|b_1 D_1 + b_2 D_2 + \cdots + b_{n+1} D_{n+1}|}{|D_1| + |D_2| + \cdots + |D_{n+1}|}$$

Sustituyendo, obtenemos:

$$\delta = \frac{|b_{n+1}D_{n+1} - D_{n+1}\sum_{i=1}^{n} b_i e_i/d_i|}{|D_{n+1}| + |D_{n+1}|\sum_{i=1}^{n}|e_i|/|d_i|}$$

$$\delta = \frac{|b_{n+1} - \sum_{i=1}^{n} b_i e_i/d_i}{1 + \sum_{i=1}^{n}|e_i|/|d_i|}$$

Se debe notar que, en la fórmula, los números e_i y d_i aparecen únicamente con la relación e_i/d_i, por lo que podemos permitir la operación de dividir cada elemento de una columna entre una constante distinta de cero, y así reducir la forma de la matriz $\mathscr{U}$ a una en la que $d_i = 1$. Obteniendo:

$$\mathscr{U}T_1 = \begin{bmatrix} 1 & 0 & \cdots & 0 & A_{n+1}^1 \\ 0 & 1 & \cdots & 0 & A_{n+1}^2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & A_{n+1}^n \\ e_1 & e_2 & \cdots & e_n & A_{n+1}^{n+1} \end{bmatrix} \tag{2.8}$$

$$\begin{array}{rcll} A_{n+1}^{n+1} & = & 1 & \\ A_{n+1}^{i} & = & -sgn\ e_i & (i = 1, \ldots, n). \end{array}$$

$$\delta = \frac{|b_{n+1} - \sum_{i=1}^{n} b_i e_i|}{1 + \sum_{i=1}^{n}|e_i|}$$

Por lo que aunque δ sea una función que depende de $A_{n+1}^1, A_{n+1}^2, \ldots, A_{n+1}^{n+1}$, hallarlo requiere aproximadamente la misma cantidad de cálculos que si estas cantidades se conocieran de antemano. Si las primeras $r < n$ filas de A fuesen linealmente independientes y las primeras $r+1$ son linealmente dependientes, no sería posible reducir $\mathscr{U}$ a la forma de la matriz (2.8) mediante operaciones elementales. Pese a ello, podríamos encontrar T_1 tal que

$$\mathscr{U}T_1 = \begin{vmatrix} 1 & 0 & \cdot & \cdot & \cdot & \cdot & 0 & A_{n+1}^1 \\ 0 & 1 & \cdot & \cdot & \cdot & \cdot & 0 & A_{n+1}^2 \\ \vdots & \vdots & \ddots & \cdot & \cdot & \cdot & \cdot & \vdots \\ 0 & 0 & \cdots & 1 & 0 & \cdots & 0 & A_{n+1}^r \\ t_1^{r+1} & t_2^{r+1} & \cdots & t_r^{r+1} & 0 & \cdots & 0 & A_{n+1}^{r+1} \\ t_1^{r+2} & t_2^{r+2} & \cdot & \cdot & \cdot & \cdot & t_n^{r+2} & A_{n+1}^{r+2} \\ \vdots & \vdots & \cdot & \cdot & \cdot & \cdot & \vdots & \vdots \\ t_1^{n+1} & t_2^{n+1} & \cdot & \cdot & \cdot & \cdot & t_n^{n+1} & A_{n+1}^{n+1} \end{vmatrix}, \ (i = 1, 2, \ldots, n). \tag{2.9}$$

Al asumir que A tiene rango n, se tiene que la matriz (t_j^i) donde $r+2 \leq i \leq n+1$, $r+1 \leq j \leq n$, no debe ser singular, por lo que, mediante el uso

de operaciones elementales en las columnas, podemos obtener de la matriz previa la siguiente:

$$\begin{bmatrix} 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & A_{n+1}^1 \\ 0 & 1 & \cdots & 0 & 0 & 0 & \cdots & 0 & A_{n+1}^2 \\ \vdots & \vdots & \ddots & 0 & 0 & 0 & \cdot & \cdot & \vdots \\ 0 & 0 & \cdots & 1 & 0 & 0 & \cdots & 0 & A_{n+1}^r \\ t_1^{r+1} & t_2^{r+1} & \cdots & t_r^{r+1} & 0 & 0 & \cdots & 0 & A_{n+1}^{r+1} \\ 0 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & A_{n+1}^{r+2} \\ 0 & 0 & \cdots & 0 & 0 & 1 & \cdots & 0 & A_{n+1}^{r+3} \\ \vdots & \vdots & \cdot & \cdot & \cdot & \cdot & \ddots & \cdot & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 1 & A_{n+1}^{n+1} \end{bmatrix} \tag{2.10}$$

Ahora sí podemos calcular el conjunto de valores A_{n+1}^i. Un conjunto de valores que se pueden usar, está dado por:

$$\begin{array}{rcll} A_{n+1}^i & = & sgn\, t_{r+1}^i & (i = 1, \ldots, r) \\ A_{r+1}^{r+1} & = & -1 & \\ A_{n+1}^i & = & 0 & (i = r+2, \ldots, n+1) \end{array}$$

Cuando tenemos (2.10), el valor de δ puede ser hallado inmediatamente resolviendo

$$\delta = \frac{|b_{r+1} - \sum_{i=1}^r b_i \; sgn \; t_{r+1}^i|}{1 + \sum_{i=1}^r |t_{r+1}^i|}$$

Si se desea una solución completa $x_1, x_2, \ldots, x_n, \delta$ hay dos maneras en las que se puede hallar. La primera, al conocer los A_{n+1}^i, podremos volver al sistema lineal original y resolver mediante los métodos convencionales. La segunda, el producto de las transformaciones elementales de las columnas que modificaban $\mathscr{U}$ hasta convertirla en la forma (2.8) o de la forma (2.10) se debe acumular. Los valores A_{n+1}^i se insertan y se vuelven a realizar operaciones elementales en las columnas, hasta llegar a la matriz identidad. El producto acumulado de todas estas transformaciones en las columnas, nos dará la inversa de $\mathscr{U}$, con lo que podremos resolver $\mathbf{x} = \mathscr{U}^{-1}\mathbf{b}$.

Al reducir $\mathscr{U}$ a las formas (2.8) o (2.10) puede ser posible alcanzar una gran precisión si en cada paso seleccionamos como pivote el elemento máximo. Esto puede hacerse mediante intercambios correctos de filas. El intercambio de los primeros n elementos de dos filas de $\mathscr{U}$ y sus correspondientes elementos en $\mathbf{b}$, es equivalente a intercambiar las ecuaciones correspondientes en el sistema de ecuaciones lineales original.

Suponiendo que el sistema (2.6) se resuelve en el sentido de Chebyshev y

junto con el requisito de que se cumpla la ecuación:

$$\sum_{j=1}^{n} A_j^i x_j - b_i = 0 \quad (i = r, \ldots, n+1;\ r \geq 2)$$

es necesario definir $A_{n+1}^i = 0$ $(i = r, \ldots, n+1)$ y resolver (2.7) como hemos explicado a lo largo del apartado. Asumiendo que las primeras n filas de A son linealmente independientes, obtenemos que la fórmula para hallar δ es

$$\delta = \frac{|b_{n+1} - \sum_{i=1}^{n} b_i e_i|}{\sum_{i=1}^{r-1} |e_i|}$$

2.2.3. Teoría en el hiperplano

Cuando $m = n+1$, veremos que los signos de los residuos son los mismos tanto para la solución minimax como para la solución mínimos cuadrados. Este resultado se extiende también a otras normas distintas de la euclídea. Para poder demostrar estas afirmaciones, debemos hacer uso de la noción del hiperplano.

Definición 2.2.1. Un hiperplano en un espacio normado E es un conjunto de puntos de la forma $H = \{x \in E :\ f(x) = c\}$, donde c es una constante y f es una forma lineal de valores reales distinta de cero (en el caso de un espacio de Banach se requerirá que sea además continua).

En el espacio $\mathbb{R}^{n+1}$ un hiperplano consistirá en todas las $(n+1)$-uplas u para las cuales $\langle u, f \rangle = c$, siendo f una $(n+1)$-upla fija y c una constante. Ahora, cuando se resuelva aproximadamente un sistema de ecuaciones:

$$\sum_{j=1}^{n} A_j^i x_j = b_i \quad (i = 1, \ldots, n+1)$$

se intenta hacer el vector:

$$r = \sum_{j=1}^{n} x_j \mathbf{A_j} - \mathbf{b}$$

tan pequeño como sea posible en la norma elegida. Los puntos de forma r permanecen en el hiperplano.

Lema 2.2.2. *Si el conjunto de vectores columna* $\{\mathbf{A_1}, \ldots, \mathbf{A_n}\}$ *es linealmente independiente en* $\mathbb{R}^{n+1}$ *y* $\mathbf{b}$ *es un elemento fijo de* $\mathbb{R}^{n+1}$*, entonces el conjunto de puntos*

$$H = \left\{ z \in \mathbb{R}^{n+1} \mid z = \sum_{j=1}^{n} x_j \mathbf{A_j} - \mathbf{b} :\ \mathbf{x_j}\ \textit{real} \right\}$$

es un hiperplano.

Definición 2.2.3. Se dice que una norma es monótona en $\mathbb{R}^n$ si tiene la propiedad de que $\|\mathbf{x}\| \leq \|\mathbf{y}\|$, siempre que los vectores $\mathbf{x} = [x_1, \ldots, x_n]$ e $\mathbf{y}= [y_1, \ldots, y_n]$ cumplan la desigualdad $|x_i| \leq |y_i|$ para $i = 1, \ldots, n$.

Todas las normas de la forma: $\|\mathbf{x}\|_\mathbf{p} = \sqrt[p]{\sum |x_i|^p}$ ($\mathbf{1} \leq \mathbf{p} \leq \infty$) tienen esta propiedad.

Teorema 2.2.4. *Sea H un hiperplano en $\mathbb{R}^n$. Los puntos de H que minimizan dos normas monótonas distintas tienen componentes que coinciden en el signo (o pueden elegirse así en el caso de no unicidad.)*

Demostración. Si $0 \in H$, el teorema es trivial. Si $0 \notin H$, tomamos $H = \{x \in \mathbb{R}^n : \langle x, u\rangle = c\}$. Si fuese necesario cambiar u por $-u$, se puede asumir que $c > 0$. Sea x un punto de H que minimiza una norma monótona $\|x\|$. Denotamos $x'_i = |x_i|\ sgn(u_i)$, se obtiene un punto x' cuyas componentes coinciden en signo con aquellas de u. Por tanto, $\langle x', u\rangle \geq \langle x, u\rangle = c > 0$, y el número $\theta = c/\langle x', u\rangle$ permanece en el intervalo $(0, 1]$. Por ser la norma monótona, se obtiene $\|\theta x'\| \leq \|x'\| \leq \|x\|$. Por $\langle \theta x', u\rangle = c$, $\theta x'$ permanece en H y minimiza la norma. □

Por el teorema precedente, se obtiene que los signos σ_i necesarios para resolver el problema minimax de $n + 1$ ecuaciones con n incógnitas pueden ser obtenidos a partir de su solución mínimos cuadrados, usando para σ_i el signo del residuo i-ésimo de dicha solución. Además, podremos determinar también el número ϵ de (2.4). El siguiente teorema resume lo hallado.

Teorema 2.2.5. *Sea y la solución de mínimos cuadrados de un sistema lineal de $n+1$ ecuaciones con n incógnitas, $r_i(x) = 0, i = 1, \ldots, n+1$. Se asume que el sistema es de rango n. Entonces la solución minimax es la solución exacta del sistema $r_i(x) = \sigma_i\epsilon$, donde $\sigma_i = sgn\ r_i(y)$ y $\epsilon = \sum r_i^2(y)/\sum |r_i(y)|$.*

Demostración. Del lema anterior, los puntos $r(x) = \sum x_j\mathbf{A_j} - \mathbf{b}$ rellenan un hiperplano $H = \{z :\ \langle z, r(y)\rangle = \langle r(y), r(y)\rangle\}$ en $\mathbb{R}^{n+1}$. Si se define el punto z con componentes $z_i = \sigma_i\epsilon$, entonces $z \in H$, ya que:

$$\langle z, r(y)\rangle = \sum z_i r_i(y) = \epsilon \sum \sigma_i r_i(y) = \epsilon \sum |r_i(y)| = \sum r_i^2(y)$$

Por otro lado, ningún punto de H está más cerca de 0 que z en la norma de Chebyshev. Para ver que z es la única solución, supongamos que existe otra solución u más pequeña que z, si $\|u\|_\infty < \|z\|_\infty$, entonces, $\langle u, r(y)\rangle = \sum u_i r_i(y) \leq (\max |u_i|) \sum |r_i(y)| < \max |z_i| \sum |r_i(y)| = \epsilon \sum |r_i(y)| = \sum r_i^2(y)$, por lo que u no pertenece a H. □

El siguiente ejemplo ilustra el resultado anterior.

Ejemplo 2.2.6. Se considera el sistema:

$$\begin{aligned} x_1 - x_2 &= 7 \\ 2x_1 + 3x_2 &= 5 \\ 3x_1 + x_2 &= -1 \end{aligned}$$

Para conseguir la solución por mínimos cuadrados, se debe minimizar la función:

$$\phi(x_1, x_2) = (x_1 - x_2 - 7)^2 + (2x_1 + 3x_2 - 5)^2 + (3x_1 + x_2 + 1)^2$$

Igualando las derivadas parciales de ϕ a cero, se obtiene:

$$\begin{aligned} 14x_1 + 8x_2 &= 14 \\ 8x_1 + 11x_2 &= 7 \end{aligned}$$

Por lo que: $x_1 = \frac{49}{45}$ y $x_2 = -\frac{7}{45}$. El vector de residuos correspondiente a esta solución de mínimos cuadrados es $(-\frac{259}{45}, -\frac{148}{45}, \frac{185}{45})$. El número ϵ es $\frac{37}{8}$. La solución minimax del sistema es, por tanto, la solución exacta del sistema:

$$\begin{aligned} x_1 - x_2 - 7 &= -\tfrac{37}{8} \\ 2x_1 + 3x_2 - 5 &= -\tfrac{37}{8} \\ 3x_1 + x_2 + 1 &= \tfrac{37}{8} \end{aligned}$$

El vector de residuos es $(-\frac{37}{8}, -\frac{37}{8}, \frac{37}{8})$. La solución minimax será $(\frac{3}{2}, -\frac{7}{8})$.

2.3. Algoritmo del ascenso

Para que los cálculos sean más fáciles, vamos a asumir que en la matriz (A^i_j) sus vectores fila satisfacen un requisito denominado la condición de Haar.

Definición 2.3.1. Un conjunto de vectores en un espacio de dimensión n se dice que satisface la condición de Haar si cada conjunto de n de ellos es linealmente independiente. Expresado de otra forma, cada selección de n vectores de tal conjunto es una base del n-espacio.

El teorema que se expone a continuación será el fundamento de un algoritmo para la solución minimax de un sistema lineal de ecuaciones. A partir de un conjunto de vectores que satisfacen la condición de Haar, es posible intercambiar un vector por otro y que se siga cumpliendo la condición-

Teorema 2.3.2 (Teorema del intercambio)**.** *Sea* $\{\mathbf{A^0}, ..., \mathbf{A^{n+1}}\}$ *un conjunto de vectores en un espacio de dimensión n y que satisface la condición de Haar. Si* 0 *pertenece a la envolvente convexa de* $\{\mathbf{A^0}, ..., \mathbf{A^n}\}$*, entonces existe un índice* $1 \leq j \leq n$ *tal que esta condición sigue siendo cierta cuando* $\mathbf{A^j}$ *se sustituye por* $\mathbf{A^{n+1}}$*.*

Demostración. Por hipótesis, existen constantes $\theta_i \geq 0$ tales que cumplen $0 = \sum_{i=0}^{n} \theta_i \mathbf{A^i}$ y $\sum_{i=0}^{n} \theta_i = 1$. La condición de Haar sería transgredida si cualquier θ_i fuera cero; por ello, $\theta_i > 0$, $i = 0, \ldots, n$. Si resolvemos es esta ecuación en $\mathbf{A^j}$, en principio con j arbitrario, obtenemos $\mathbf{A^j} = \sum_{\substack{i=0 \\ i \neq j}}^{\mathbf{n}} \frac{-\theta_\mathbf{i}}{\theta_\mathbf{j}} \mathbf{A^i}$ para cualquier $1 \leq j \leq n$. Debido a que $\{\mathbf{A^0}, ..., \mathbf{A^n}\}$ genera un espacio de dimensión n, es posible escribir además $\mathbf{A^{n+1}} = \sum_{\mathbf{i=0}}^{\mathbf{n}} \lambda_\mathbf{i} \mathbf{A^i}$ para ciertos λ_i. Luego,

$$\begin{aligned}
0 &= \mathbf{A^{n+1}} - \sum_{\mathbf{i=0}}^{\mathbf{n}} \lambda_\mathbf{i} \mathbf{A^i} \\
&= \mathbf{A^{n+1}} - \lambda_\mathbf{j} \mathbf{A^j} - \sum_{\substack{i=0 \\ i \neq j}}^{\mathbf{n}} \lambda_\mathbf{i} \mathbf{A^i} \\
&= \mathbf{A^{n+1}} - \lambda_\mathbf{j} \sum_{\substack{i=0 \\ i \neq j}}^{\mathbf{n}} \frac{-\theta_\mathbf{i}}{\theta_\mathbf{j}} \mathbf{A^i} - \sum_{\substack{i=0 \\ i \neq j}}^{\mathbf{n}} \lambda_\mathbf{i} \mathbf{A^i} \\
&= \mathbf{A^{n+1}} + \sum_{\substack{i=0 \\ i \neq j}}^{\mathbf{n}} \left(\frac{\lambda_\mathbf{j} \theta_\mathbf{i}}{\theta_\mathbf{j}} - \lambda_\mathbf{i} \right) \mathbf{A^i}
\end{aligned}$$

Ahora, si j se selecciona para que $\lambda_j \theta_i / \theta_j - \lambda_i \geq 0$, $i = 0, \ldots, n$, $i \neq j$, entonces la ecuación final expresa 0 como una combinación lineal no negativa de $\mathbf{A^0}, ..., \mathbf{A^{n+1}}$ en la que $\mathbf{A^j}$ no aparece, lo que es suficiente para probar que 0 se encuentra en la envolvente convexa de estos puntos. El requisito en j es que $\lambda_j / \theta_j \geq \lambda_i / \theta_i$, $i \neq j$; en otras palabras, se debe seleccionar j para que λ_j / θ_j sea la mayor de las ratios λ_i / θ_i. El índice j será único, ya que, si hubiera dos λ_j / θ_j donde se alcanza el máximo, entonces uno de los coeficientes de la ecuación superior desaparecería, contradiciendo la condición de Haar. □

Primero, se va a describir cómo funciona el algoritmo en un sistema de ecuaciones con una única incógnita para familiarizarse con el proceso. Se deberá resolver el problema Tchebycheff asociado al sistema de ecuaciones $a_i x = b_i$ $(i = 1, \ldots, m)$. En cada paso de este algoritmo, se tiene un punto x_0 y un par de índices j y k tales que $r_j(x_0) = r_k(x_0)$, $a_j \leq 0 \leq a_k$, y

$a_j \neq a_k$. Bajo estas circunstancias, es fácil ver que x_0 es un punto mínimo de la función máx $\{r_j(x), r_k(x)\}$. Se selecciona i tal que $r_i(x_0) = \delta(x_0)$. Si $a_i < 0$, se procede a la intersección de r_i y r_k. Si $a_i > 0$, se procede a la intersección de r_i y r_j. Si $a_i = 0$, se procede a la intersección de r_i con r_j o con r_k con tal de que tengan coeficiente de x, a_j o a_k, no nulo. En el primero de los tres casos, por ejemplo, se reemplaza x_0 por el punto nuevo $x = (b_k - b_i)/(a_k - a_i)$. Se reemplazaría también j por i y se empezaría de nuevo.

La descripción del algoritmo para una dimensión cualquiera es la siguiente. Hay que buscar un punto donde la función

$$\Delta(x) = \max_{1 \leq i \leq m} |r_i(x)| = \max_{1 \leq i \leq m} \left| \left\langle A^i, x \right\rangle - b_i \right|$$

alcance su valor mínimo. Se asume que la condición de Haar se satisface por el conjunto de vectores $\{\mathbf{A^1}, ..., \mathbf{A^m}\}$. La idea básica del algoritmo es calcular la solución minimax de una sucesión de subsistemas, cada uno de los cuales se compone de $n+1$ ecuaciones. Por el teorema 2.1.4, la solución de uno de los subsistemas es el punto deseado. Por otro lado, sólo puede haber un número finito de estos subsistemas y esta observación es la base para la prueba de que el algoritmo termina en un número finito de pasos.

En cada ciclo de la computación, tendremos un conjunto de $n+1$ índices $J = \{i_0, \ldots, i_n\}$ y un vector de signos $\sigma = \{\sigma_0, \ldots, \sigma_n\}$, tal que

$$0 \in \mathscr{H}\left\{\sigma_0 \mathbf{A^{i_0}}, \ldots, \sigma_\mathbf{n} \mathbf{A^{i_n}}\right\} \tag{2.11}$$

Se resuelve el siguiente sistema de $n+1$ ecuaciones lineales para determinar el vector $\mathbf{y} = (\mathbf{y_1}, \ldots, \mathbf{y_n})$ y un número e:

$$\sigma_j r_{i_j}(y) = e \quad (j = 0, \ldots, n) \tag{2.12}$$

Sin pérdida de generalidad, podemos asegurar que $e > 0$, pues en caso contrario se pueden cambiar los signos de todos σ_j sin perder la propiedad de que 0 esté en la envolvente convexa $\mathscr{H}\left\{\sigma_0 \mathbf{A^{i_0}}, \ldots, \sigma_\mathbf{n} \mathbf{A^{i_n}}\right\}$. Las condiciones (2.11) y (2.12) implican que $\mathbf{y}$ es la solución minimax del sistema:

$$\left\langle A^{i_j}, \mathbf{y} \right\rangle = b_{i_j} \quad (j = 0, \ldots, n)$$

Si $e = \Delta(y)$, entonces por el teorema 2.1.2, $\mathbf{y}$ es una solución minimax del sistema original de m ecuaciones. En caso contrario, existe al menos, un índice α (que no se encuentra en J) tal que $|r_\alpha(\mathbf{y})| = \Delta(\mathbf{y})$, (basta sin embargo elegir α para que $|r_\alpha(y)| > e$.). Ahora, sea $\mu = sgn\ r_\alpha(y)$. Usando el teorema de intercambio, se reemplaza uno de los vectores $\sigma_0 \mathbf{A^{i_0}}, \ldots, \sigma_\mathbf{n} \mathbf{A^{i_n}}$ por $\mu \mathbf{A}^\alpha$, de tal manera que el origen sigue estando en la envolvente convexa

del nuevo conjunto; esto es gracias a la propiedad (2.11). Nos encontramos en la misma situación que al inicio del ciclo y se repite el procedimiento de la misma forma.

La necesidad de calcular $\mathbf{y}$ y e implica hacer algunas conjeturas sobre los datos dados, una conveniente sobre la matriz

$$\begin{bmatrix} \sigma_0 & A_1^{i_0} & \cdots & A_n^{i_0} \\ \vdots & \vdots & \cdots & \vdots \\ \sigma_n & A_1^{i_n} & \cdots & A_n^{i_n} \end{bmatrix}$$

es que debe ser no singular.

Los cálculos del algoritmo pararán solo cuando $e = \Delta(y)$, y esto implica que $\mathbf{y}$ es una solución. Solo existe un número finito de conjuntos J. Faltaría demostrar que los cálculos no son "cíclicos", es decir, que no devuelven infinitamente muchas veces el mismo subconjunto J. Bastará demostrar que el número e es una función de J que aumenta estrictamente en cada iteración.

Para tal fin, suponemos para simplificar la notación que en un cierto paso J es el conjunto de índices $\{1, \ldots, n+1\}$ y que α es el índice $n+2$. Suponemos, además, que en el siguiente paso J será $J' = \{2, \ldots, n+2\}$. Sean $\mathbf{y}'$ y e' los valores de $\mathbf{y}$ y e que corresponden al nuevo conjunto J'. Por elección de α, $|r_2(y)| < |r_{n+2}(y)|$, mientras $|r_2(y')| = |r_{n+2}(y')|$. Por tanto, $\mathbf{y} - \mathbf{y}' \neq \mathbf{0}$. Puesto que $\langle \sigma_i \mathbf{A^i}, \mathbf{y} - \mathbf{y}' \rangle = \sigma_i r_i(y) - \sigma_i r_i(y') = e - e'$ para $i = 2, \ldots, n+1$, la condición de Haar implica que $e - e' \neq 0$. Ahora,

$$\langle \sigma_{n+2} \mathbf{A^{n+2}}, \mathbf{y} - \mathbf{y}' \rangle = \sigma_{\mathbf{n+2}} \mathbf{r_{n+2}}(\mathbf{y}) - \sigma_{\mathbf{n+2}} \mathbf{r_{n+2}}(\mathbf{y}') > \mathbf{e} - \mathbf{e}'.$$

Si $e - e' > 0$, entonces $\langle \sigma_i \mathbf{A^i}, \mathbf{y} - \mathbf{y}' \rangle > \mathbf{0}$, $i = 2, \ldots, n+2$, contradiciendo que $0 \in \mathscr{H}\left\{\sigma_i \mathbf{A^i} : \mathbf{2} \leq \mathbf{i} \leq \mathbf{n+2}\right\}$. Por tanto, $e - e' < 0$, que prueba el carácter estrictamente creciente de la magnitud del error e.

El conjunto inicial J puede ser tomado arbitrariamente, y se debe encontrar entonces una solución no trivial del sistema

$$\sum_{j=0}^{n} \theta_j \mathbf{A^{i_j}} = \mathbf{0},$$

y se toma $\sigma_j = sgn\ \theta_j$.

Un método para organizar los cálculos es el siguiente. Por la propiedad (2.12): $e = \sigma_j r_{i_j}(y) = \sigma_j \left[\left\langle \mathbf{A^{i_j}}, \mathbf{y} \right\rangle - b_{i_j}\right]$, $j = 0, \ldots, n$, puede reescribirse como el sistema lineal de ecuaciones $-\sigma_j e + \left\langle \mathbf{A^{i_j}}, \mathbf{y} \right\rangle = b_{i_j}$, que en forma

matricial se expresa:

$$\begin{bmatrix} \sigma_0 & A_1^{i_0} & \cdots & A_n^{i_0} \\ \vdots & \vdots & \cdots & \vdots \\ \sigma_n & A_1^{i_n} & \cdots & A_n^{i_n} \end{bmatrix} \begin{bmatrix} -e \\ y_1 \\ . \\ . \\ . \\ y_n \end{bmatrix} = \begin{bmatrix} b_{i_0} \\ . \\ . \\ . \\ b_{i_n} \end{bmatrix} \tag{2.13}$$

Si asumimos que la matriz A_J de este sistema tiene una inversa $C = (C_j^i)$, entonces se puede escribir:

$$\begin{bmatrix} -e \\ y_1 \\ . \\ . \\ . \\ y_n \end{bmatrix} = \begin{bmatrix} C_0^0 & \cdots & C_n^0 \\ \vdots & \cdots & \vdots \\ C_0^n & \cdots & C_n^n \end{bmatrix} \begin{bmatrix} b_{i_0} \\ . \\ . \\ . \\ b_{i_n} \end{bmatrix}$$

Ya que C es la inversa de A_J, se tiene que:

$$\begin{bmatrix} C_0^0 & \cdots & C_n^0 \\ \vdots & \cdots & \vdots \\ C_0^n & \cdots & C_n^n \end{bmatrix} \begin{bmatrix} \sigma_0 & A_1^{i_0} & \cdots & A_n^{i_0} \\ \vdots & \vdots & \cdots & \vdots \\ \sigma_n & A_1^{i_n} & \cdots & A_n^{i_n} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & 0 & \ddots & \vdots \\ 0 & \cdots & \cdots & 1 \end{bmatrix}$$

De aquí es evidente obtener que $\sum_{j=0}^n \sigma_j C_j^0 = 1$ y $\sum_{j=0}^n C_j^0 A^{i_j} = 0$. Por lo tanto, los números $\sigma_j C_j^0$ son los coeficientes necesarios para expresar el hecho de que 0 permanece a la envolvente convexa de los puntos $\sigma_j A^{i_j}$. Estos coeficientes entran en los cálculos relacionados con el teorema de intercambio 2.3.2. De la prueba del teorema, se observa que se debe expresar $\mu \mathbf{A}^\alpha$ como una combinación lineal de $\sigma_0 \mathbf{A^{i_0}}, \cdots, \sigma_\mathbf{n} \mathbf{A^{i_n}}$. Si se fija $\mathbf{A}^\alpha = \sum_{\mathbf{j=0}}^{\mathbf{n}} \lambda_\mathbf{j} \mathbf{A^{i_j}}$, entonces los coeficientes λ_j pueden ser obtenidos resolviendo el sistema lineal

$$(\lambda_0, \cdots, \lambda_n) \begin{bmatrix} \sigma_0 & A_1^{i_0} & \cdots & A_n^{i_0} \\ \vdots & \vdots & \cdots & \vdots \\ \sigma_n & A_1^{i_n} & \cdots & A_n^{i_n} \end{bmatrix} = (\mu, A_1^\alpha, \cdots, A_n^\alpha)$$

La solución viene dada por

$$(\lambda_0, \cdots, \lambda_n) = (\mu, A_1^\alpha, \cdots, A_n^\alpha) \begin{bmatrix} C_0^0 & \cdots & C_n^0 \\ \vdots & \cdots & \vdots \\ C_0^n & \cdots & C_n^n \end{bmatrix}$$

Como $\mu\mathbf{A}^{\alpha} = \sum_{\mathbf{j=0}}^{\mathbf{n}} (\mu\sigma_{\mathbf{j}}\lambda_{\mathbf{j}}) \left(\sigma_{\mathbf{j}}\mathbf{A}^{\mathbf{i_j}}\right)$, los índices que deben ser calculados en el teorema de intercambio 2.3.2 son $\mu\sigma_j\lambda_j/\sigma_j C_j^0 \equiv \mu\lambda_j/C_j^0$. El número β es elegido como el mayor índice que cumple lo anterior.

Es conveniente observar que en el proceso de cambio de un ciclo al siguiente la matriz A_J sólo cambia en una fila. El efecto de este cambio en el cálculo de la inversa $C = A_J^{-1}$ se describe en el siguiente teorema.

Teorema 2.3.3. *Sea A una matriz no singular $n \times n$ y $C_1, \cdots, C_n$ las columnas de su inversa $A^{-1} = [C_1, \ldots, C_n]$. Sea $\hat{A}$ la matriz obtenida reemplazando la fila β de A por un vector v. Si $\lambda \equiv \langle v, C_\beta \rangle \neq 0$, entonces $\hat{A}$ es no singular, y las columnas de su inversa están dadas por las fórmula $\hat{C}_\beta = \lambda^{-1} C_\beta$ y $\hat{C}_j = C_j - \langle v, C_j \rangle \hat{C}_\beta$, para $j \neq \beta$.*

Demostración. Para verificar que $\hat{A}\hat{C} = I$, es necesario calcular el producto interno de $\hat{A}^i$, esto es, la i-ésima fila de $\hat{A}$, con $\hat{C}_j$. Hay cuatro casos. En el caso 1, $i = \beta$, por consiguiente, $\left\langle \hat{A}^\beta, \hat{C}_\beta \right\rangle = \langle v, \lambda^{-1} C_\beta \rangle = \lambda^{-1} \langle v, C_\beta \rangle = 1$.
En el caso 2, $i \neq \beta$ y $j = \beta$. Luego, $\left\langle \hat{A}^i, \hat{C}_\beta \right\rangle = \langle A^i, \lambda^{-1} C_\beta \rangle = \lambda^{-1} \langle A^i, C_\beta \rangle = 0$.
En el caso 3. $i = \beta$ y $j \neq \beta$. Por lo tanto, $\left\langle \hat{A}^i, \hat{C}_j \right\rangle = \left\langle v, C_j - \langle v, C_j \rangle \tilde{\mathrm{C}}_\beta \right\rangle = \langle v, C_j \rangle - \langle v, C_j \rangle \lambda^{-1} \langle v, C_\beta \rangle = 0$.
En el caso 4, $i \neq \beta$ y $j \neq \beta$. Así pues, $\left\langle \hat{A}^i, \hat{C}_j \right\rangle = \left\langle A^i, C_j - \langle v, C_j \rangle \hat{C}_\beta \right\rangle = \langle A^i, C_j \rangle - \langle v, C_j \rangle \lambda^{-1} \langle A^i, C_\beta \rangle = \langle A^i, C_j \rangle = \delta_{ij}$. □

En el algoritmo, se debe reemplazar la fila $(\sigma_\beta, A_1^{i_\beta}, \ldots, A_n^{i_\beta})$ por una nueva fila $(\mu, A_1^\alpha, \ldots, A_n^\alpha)$. Es necesario el número λ que es el producto interior de la nueva fila con la β-ésima columna de C; en concreto, $\mu C^0 + \sum_{j=1}^n A_j^\alpha C_\beta^j$, pero este es el número λ_β previamente computado. En el programa se simplificará la notación denotando con y_0 el número $-e$.

2.3.1. Código en MATLAB

```
1  function [x, iter] = algoritmoascendente2( A, b, J)
2
3  % Queremos que nos de la aproximación que es x
4  % Pedimos :
5  %   A es la matriz de coeficientes
6  %   b es Ax=B
7  %   J filas que queremos
8  % No tenemos que comprobar la condición de Haar. Se
       supone que nos dan la
9  % matriz bien
```

```
10
11      % Creamos una matriz A con las filas que nos han dado
            con J
12      [m,n] = size(A);
13      A_J = zeros(length(J),n);
14      b_J = zeros(length(J),1);
15      A_J = A(J(:),:);
16      b_J = b(J(:),:);
17
18      % Debemos resolver el sumatorio
19      S = [A_J ones(n+1,1)];
20      c = [zeros(n,1);1];
21      theta = c'/S;
22      if any(isnan(theta))
23      error('Theta tiene componentes NaN')
24      end
25
26      % Vemos los signos de theta
27      sgn = sign(theta);
28
29      % Pasamos al paso 4
30      C = inv([sgn',A_J]);
31
32      % elegimos los b correspondientes y resolvemos el
            sumatorio
33      y = C*b_J;
34
35      % queremos hallar los restos
36      e = y(1);
37      y = y(2:length(J));
38      R = A*y-b;
39      [amax,alpha] = max(abs(R)); % seleccionamos el valor
            maximo y el indice donde se encuentra
40      disp(y)
41      disp(R)
42      disp(J)
43      disp(alpha)
44
45      iter = 1;
46
47      while abs(amax - abs(e)) > 2^-30
48
49      % pasamos al caso en que a es distinto de y_0
50      % cogemos el signo de alpha y le denominamos mu
51      mu = sign(R(alpha));
52
53      % buscamos ahora los lambdas
54      lambda = mu*C(1,:) + A(alpha,:)*C(2:length(J),:);
55
```

```
56        % seleccionamos beta para que sea maximo
57        [bmax,beta] = max(mu*lambda./C(1,:));
58
59        % paso 4
60        C(:,beta) = C(:,beta)/lambda(beta);
61
62        for i = 1:length(J)
63        if i ~= beta
64        C(:,i) = C(:,i)-lambda(i)*C(:,beta);
65        end
66        end
67        J(beta) = alpha;
68
69        % renuevo b y A
70        b_J = b(J(:));
71        A_J = A(J(:),:);
72
73        % aquí es igual que antes
74        y = C*b_J;
75
76        % queremos hallar los restos
77        e = y(1);
78        y = y(2:length(J));
79        R = A*y-b;
80        [amax,alpha] = max(abs(R)); % seleccionamos el valor
              maximo y el indice donde se encuentra
81        disp(y)
82        disp(R)
83        disp(J)
84        disp(alpha)
85        iter = iter+1;
86        end
87        x = y;
88        end
```

2.3.2. Método ascendente con descomposición LU

Este algoritmo consiste en empezar con un subsistema de referencia y modificar en cada paso una ecuación. Cada modificación se llevará a cabo de la siguiente forma:

Podemos asumir que $\mathbf{A_1}, \ldots, \mathbf{A_{n+1}}$ es el conjunto de la referencia. Sea $\mathbf{x} = (x_1, \ldots, x_n)$ una solución de Chebyshev del correspondiente subsistema de referencia, por lo que tenemos $\epsilon, \lambda_1, \ldots, \lambda_{n+1}$ que satisfacen:

- $\sum_{i=1}^{n+1} \lambda_i \mathbf{A_i} = 0$
- $\epsilon = -\sum_{i=1}^{n+1} \lambda_i b_i / \sum_{j=1}^{n+1} |\lambda_j|$

- $r_i(x) = \sigma_i \epsilon, \quad i = 1, \ldots, n+1 \quad \sigma_i = sgn\ (\lambda_i)$

Si $\mathbf{x}$ no es una solución de Chebyshev para el sistema completo, existirá un $\alpha \in [n+2, \ldots, m]$ para el que $|r_\alpha(x)| > |\epsilon|$. Sean $\rho_1, \ldots, \rho_{n+1}$ escalares para los cuales

$$\mathbf{A}_\alpha = \sum_{\mathbf{i=1}}^{\mathbf{n+1}} \rho_\mathbf{i} \mathbf{A_i}$$

Imponemos las siguientes condiciones:

- *Condición 1.* $\lambda_i \neq 0$ para todo $i = 1, \ldots, n+1$. Si se cumple esto, sea $\beta \in [1, \ldots, n+1]$ tal que

$$\frac{\sigma_\alpha \sigma_\epsilon \rho_\beta}{\lambda_\beta} = \max_{1 \leq i \leq n+1} \frac{\sigma_\alpha \sigma_\epsilon \rho_i}{\lambda_i}$$

  donde $\sigma_\alpha = sgn\ (r_\alpha(x))$, y $\sigma_\epsilon = sgn\ (\epsilon)$.

- *Condición 2.* $\mathbf{A_1}, \ldots, \mathbf{A}_{\beta-\mathbf{1}}, \mathbf{A}_{\beta+\mathbf{1}}, \ldots, \mathbf{A_{n+1}}, \mathbf{A}_\alpha$ es un conjunto de referencias.

Resolvemos para poder obtener una solución Chebyshev $\mathbf{x}' = (x'_1, \ldots, x'_n)$ para el susbsistema de referencia:

$$\begin{bmatrix} \mathbf{A_1} \\ \vdots \\ \mathbf{A}_{\beta-\mathbf{1}} \\ \mathbf{A}_{\beta+\mathbf{1}} \\ \vdots \\ \mathbf{A_{n+1}} \\ \mathbf{A}_\alpha \end{bmatrix} \begin{bmatrix} z_1 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ z_n \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ b_{\beta-1} \\ b_{\beta+1} \\ \vdots \\ b_{n+1} \\ b_\alpha \end{bmatrix}$$

generando $\epsilon', \lambda'_1, \ldots, \lambda'_{\beta-1}, \lambda'_{\beta+1}, \lambda'_{n+1}, \lambda'_\alpha$ tales que

- $\sum_{\substack{i=1 \\ i \neq \beta}}^{n+1} \lambda'_i \mathbf{A_i} + \lambda'_\alpha \mathbf{A}_\alpha = \mathbf{0}$

- $\epsilon' = -\dfrac{\sum_{\substack{i=1 \\ i \neq \beta}}^{n+1} \lambda'_i b_i + \lambda'_\alpha b_\alpha}{|\lambda'_\alpha| + \sum_{\substack{j=1 \\ j \neq \beta}}^{n+1} |\lambda'_j|}$

- $r_i(x') = \sigma'_i \epsilon', \ \ i = 1, \ldots, \beta-1, \beta+1, \ldots, n+1, \alpha$ y con $\sigma'_i = sgn(\lambda'_i)$

Así que obtenemos,

$$\lambda_i' = \sigma_\alpha \lambda_\alpha' \lambda_i \left[\frac{\rho_\beta \sigma_\alpha}{\lambda_\beta} - \frac{\rho_i \sigma_\alpha}{\lambda_i} \right], \quad i = 1, \ldots, n+1; i \neq \beta$$

Además, si $K = |\lambda_\alpha'| + \sum_{\substack{k=1 \\ k \neq \beta}}^{n+1} |\lambda_k'|$ y $c = \frac{|\lambda_\alpha'|}{K}$, se puede ver que

$$|\epsilon'| = c|r_\alpha(x)| + (1-c)|\epsilon|.$$

Es importante notar que si la condición 1 se satisface en el segundo conjunto de referencia (esto es, $\lambda_i' \neq 0$, para $i = 1, \ldots, \beta-1, \beta+1, n+1, \alpha$), entonces $c > 0$. Por lo tanto, $|\epsilon'| > |\epsilon|$ ya que $|r_\alpha(x)| > |\epsilon|$. La desigualdad estricta $|\epsilon'| > |\epsilon|$ implica, por un simple argumento de contradicción, que, si se elige una referencia inicial y se modifica como arriba mediante intercambio sucesivo de filas de la matriz A por filas del conjunto de referencia, y, si se mantienen las condiciones 1 y 2, entonces el proceso converge a una solución de Chebyshev.

Se va a llevar a cabo la eliminación de Jordan, esto es, dados índices $\{i_1, \ldots, i_n\} \subseteq \{1, \ldots, m\}$, números $\lambda_1, \ldots, \lambda_{n+1}$, se cumple:

$$\sum_{k=1}^{n+1} \lambda_k = 1$$

$$\sum_{k=1}^{n+1} \lambda_k \mathbf{A_k} = \mathbf{0}.$$

Fijando $\sigma_k = sgn\ (\lambda_k)$ para $k = 1, \ldots, n+1$, se forma la siguiente matriz mediante $n+1$ operaciones con pivotes:

$$C = \begin{bmatrix} A_{i_1}^T & \cdots & A_{i_{n+1}}^T \\ \sigma_1 & \cdots & \sigma_{n+1} \end{bmatrix}^{-1}$$

Cada paso en que se intercambia una fila, se lleva a cabo la siguiente operación:

$$[x_1, \ldots, x_n, \epsilon] = [d_{i_1}, \ldots, d_{i_{n+1}}]C$$

Calculamos

$$r_j = \sum_{k=1}^{n+1} A_k^j x_k - d_j \ \text{ para todo } j \neq i_1, \ldots, i_{n+1},$$

seleccionamos α tal que $|r_\alpha| =$ máx formando

$$[\rho_1, \ldots, \rho_{n+1}] = [A_1^\alpha, A_2^\alpha, \ldots, A_n^\alpha, sgn\ (r_\alpha)]C^T$$

La última columna de C es de la forma:

$$\begin{bmatrix} \lambda_1/G \\ \lambda_2/G \\ \vdots \\ \lambda_{n+1}/G \end{bmatrix}$$

donde $G = \sum_{k=1}^{n+1} |\lambda_k|$. β es seleccionado como el índice para el que se cumple que $sgn\ (r_\alpha) sgn(\epsilon)\rho_\beta / C_{n+1}^{\beta}$ es máximo. Una elección adecuada del pivote de C, terminaría con el paso del intercambio.

Mientras que los intercambios de filas y columnas pueden permitirse durante la secuencia inicial de pasos de eliminación de Jordan que forman C, de modo que se puedan seleccionar los elementos pivote de mayor magnitud posible, no es posible la elección del pivote durante las actualizaciones posteriores de C.

Para la resolución del método es necesario llevar a cabo una descomposición LU.

Empezando desde cualquier subsistema de referencia de uno de los dados por el sistema, el método de intercambio produce un nuevo subsistema de referencia con el coste de resolver tres conjuntos de $n+1$ ecuaciones lineales. Estos son:

1. $P\lambda = q_1$
2. $P^T x = q_2$
3. $P\rho = q_3$

Si los tres sistemas de ecuaciones previos fuesen dados de manera aislada, el método general para resolverlos sería hacer una descomposición LU adecuada de P, usando la eliminación gaussiana. Sin embargo, con el algoritmo de Stiefel, lo que se intenta es que, como la matriz P' proviene de la matriz P con un cambio en la columna β-ésima, la descomposición LU de P' es idéntica en ciertas partes a la descomposición LU de P.

Procedemos a explicar cómo se llevaría a cabo esta descomposición LU.

1. Se seleccionan $n+1$ índices $\{i_1, \ldots, i_{n+1}\} \subseteq \{1, \ldots, m\}$ tal que la matriz

$$P = \begin{bmatrix} \mathbf{A_{i_1}} & d_{i_1} \\ \vdots & \vdots \\ \mathbf{A_{i_{n+1}}} & d_{i_{n+1}} \end{bmatrix} \quad \text{con} \quad P^T = \begin{bmatrix} \mathbf{A_{i_1}^T} & \cdots & \mathbf{A_{i_{n+1}}^T} \\ d_{i_1} & \cdots & d_{i_{n+1}} \end{bmatrix}$$

no es singular. Si esto no se puede llevar a cabo, el programa acabaría con una indicación. El usuario debería luego comprobar si el sistema $A\mathbf{x} = \mathbf{b}$ se puede resolver de manera exacta.

2. Se lleva a cabo la eliminación gaussiana en P^T hasta obtener un producto de una matriz triangular inferior L y una matriz triangular superior U. En cada columna, el elemento de mayor magnitud va a ser usado como pivote. Si la descomposición LU de una matriz difiere de P^T en únicamente la columna β-ésima, el programa puede ser más eficiente puesto que podremos ahorrar el calcular las primeras $\beta - 1$ columnas. Si el rango de $P^T < n + 1$, se termina.

3. Resolvemos

$$P^T \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_{n+1} \end{bmatrix} = LU \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_{n+1} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ -1 \end{bmatrix}$$

para que así, λ_i satisfaga $\sum_{i=1}^{n+1} \lambda_i \mathbf{A_i} = \mathbf{0}$ y se cumpla $\sum_{i=1}^{n+1} \lambda_i b_i = -1$. Si cualquiera de los λ_i es igual a 0, se termina.

4. Fijamos $\epsilon = 1/\sum_{i=1}^{n+1} |\lambda_i|$. Si ϵ es menor que cualquier valor de ϵ previamente calculado, pasamos directamente al paso 9.

5. Resolvemos

$$P \begin{bmatrix} x_1 \\ \vdots \\ x_{n+1} \end{bmatrix} = \epsilon \begin{bmatrix} sgn\ (\lambda_1) \\ \vdots \\ sgn\ (\lambda_{n+1}) \end{bmatrix}$$

x_{n+1} resultará siendo -1.

6. Calculamos $r_j(x) = \sum_{k=1}^{n} A_k^j x_k - b_j$ para cada $j \notin [i_1, \ldots, i_{n+1}]$. Sea α el índice en que $|r_\alpha(\mathbf{x})|$ es máximo. Si $|r_\alpha(\mathbf{x})| \leq \epsilon$, entonces $(x_1, \ldots, x_n)$ es candidato a ser solución de Chebyshev de $A\mathbf{x} = \mathbf{b}$, pasaríamos al paso 10.

7. Resolvemos

$$P^T \mu = \begin{bmatrix} \mathbf{A_\alpha^T} \\ b_\alpha \end{bmatrix}.$$

8. Encontramos $\beta \in \{1, \ldots, n+1\}$ tal que $(\mu_\beta/\lambda_\beta) sgn\ (r_\alpha(x))$ es máximo. Reemplazamos el conjunto de índices $\{i_1, \ldots, i_{n+1}\}$ por

$$\{i_1, \ldots, i_{\beta-1}, \alpha, i_{\beta+1}, \ldots, i_{n+1}\}.$$

Reemplazamos la β-ésima columna de P^T por $\begin{bmatrix} A_\alpha^T \\ b_\alpha \end{bmatrix}$. Volvemos al paso 2.

9. Recuperamos el conjunto anterior de índices $\{i_1, \ldots, i_{n+1}\}$ y también recuperamos la descomposición LU anterior.

10. Iterativamente, perfeccionamos la solución del sistema

$$P \begin{bmatrix} x_1 \\ \ldots \\ x_{n+1} \end{bmatrix} = \epsilon \begin{bmatrix} sgn\ (\lambda_1) \\ \vdots \\ sgn\ (\lambda_{n+1}) \end{bmatrix}$$

Comprobamos los residuos $r_j(x)$ para $j \notin \{i_1, \ldots, i_{n+1}\}$. Si $|r_\alpha(x)| = \text{máx}_j\, |r_j(x)| \leq \epsilon$, entonces, tendremos $(x_1, \ldots, x_n)$ como la solución Chebyshev. Si esta comprobación de los residuos no se cumpliera, pero se hubiera llevado a cabo todo el proceso y el último valor de ϵ es mayor que el actual valor de ϵ, volvemos a los valores anteriores de $x_1, \ldots, x_n$ como una solución dudosa. Si no, se vuelve al paso 7.

2.3.3. Código en MATLAB

```
1
2       function [x, refset, epsilon, iter] = alg_bargol4( A, b
            )
3
4       % algoritmo de ascenso descrito por Bartels y Golub
5       % algoritmo que resuelve en sentido Chebyshev un sistema
            de ecuaciones
6       % lineales Ax = b
7
8       % Parámetros de la función:
9       % A : matriz m x n de coeficientes
10      % b : vector columna m x 1
11      % x : vector solución del sistema n x 1
12      % refset : números de las ecuaciones en la referencia
            final
13      % epsilon : norma infinito del residuo en la solución
14      % iter : número de iteraciones
15
16      %m : número de ecuaciones
17      %n : número de incógnitas
18
19      % Inicializacion de arrays auxiliares
20
```

```
21        [m,n] = size(A);
22        r = [1:n+1];   %permutación de filas
23        ix = [1:m]   ; %permutación de columnas. Las primeras n
              +1 columnas tras
24        %la eliminación fijan la referencia inicial
25        lambda = zeros(n+1,1);
26        w = zeros(n+1,1);
27        x = zeros(n,1);
28        sv = zeros(n+1,1); sv(n+1) = -1.0d0;
29        lasteps = -1.0; %última norma calculada del residuo
30        iter = 0;
31        xr = zeros(n+1,1);
32
33        %Se debe determinar una referencia inicial :
              subconjunto de n+1
34        %ecuaciones de A que determinan la soución minimax del
              problema.
35
36        %La referencia incial del subsistema es elegida
              haciendo una copia de
37        %la transpuesta de A y en la última fila poniendo el
              vector b y
38        %llevando a cabo una eliminación gaussiana con
              intercambios de filas
39        %y columnas para seleccionar el mayor pivote posible.
40
41        TAB = [A'; b'];
42
43        %Iniciamos la eliminación gaussiana en TAB con pivotaje
               total.
44
45        for k = 1:n+1      %etapas de la eliminacion gaussiana (
              por columnas)
46        apiv = 0.0d0;   %módulo del elemento en TAB(k:n+1,k:m)
              de mayor módulo
47        for i = k:n+1
48        [amax,jmax] = max(abs(TAB(r(i),ix(k:m))));
49        if (amax > apiv)
50        apiv = amax; %valor máximo
51        imax = i; %fila del valor máximo
52        ipivot = r(imax); jpivot = ix(k+jmax-1);
53        pivot = TAB(ipivot,jpivot);
54        end
55        end
56
57        if (apiv < eps)
58        refset = ix;
59        error('Matriz singular. Rango insuficiente.')
60        end
```

```
61
62        iaux = r(imax);
63        r(imax) = r(k);
64        r(k) = iaux;
65        ipivot = iaux;
66
67        jaux = ix(k+jmax-1);
68        ix(k+jmax-1) = ix(k);
69        ix(k) = jaux;
70        jpivot = jaux;
71
72        % Eliminacion columna a columna
73
74        for j = k+1 : m
75        jcol = ix(j);
76        mult = TAB(ipivot,jcol)/pivot; % multiplicadores.
              cocientes fila
77        TAB(ipivot,jcol) = mult;       % pivotal entre pivot
78        for i = k+1 : n+1
79        irow = r(i);
80        TAB(irow,jcol) = TAB(irow,jcol) - TAB(irow,jpivot)*mult;
81        end
82        end
83        end
84
85        % Fin de la eliminación gaussiana
86
87        %P es la matriz del subsistema de A asociado a cada
              referencia (matriz
88        % ampliada del subsistema (n+1)x(n+1) de Ax = b cuyos í
              ndices de fila
89        % están definidos por el array ix(:)).
90
91        refset = ix(1:n+1);
92        P = [A(ix(1:n+1), :), b(ix(1:n+1))];
93
94        done = false;
95        bb  = 1; % se factoriza por columnas P, iniciando la
              factorización con
96        a1 = 1;          % la primera columna como columna
              pivotal.
97
98        while (~done)
99
100       % el siguiente apartado del programa lleva a cabo una
              reducción
101       % gaussiana mediante operaciones en las columnas de la
              matriz asociada
```

```
102         % con las ecuaciones de la referencia, formando una
                matriz triangular
103         % superior e inferior. Cada elemento de la diagonal de
                la matriz
104         % inferior será 1. Debemos intercambiar las filas para
                que el mayor
105         % pivote de cad columna se use. Se asume que las b-1
                columnas ya han
106         % sido calculadas. Si la matriz no tiene rango máximo,
                se deja a
107         % decisión del usuario determinar si el sistema dado
                puede ser resuelto
108         % de manera exacta.
109
110         %FACTORIZACIÓN P^T. Etapa factorización por columnas (
                desde columna
111         % bb) Cuando se entra con bb>1 se evita la recomputación
                 de la
112         % factorizacion para columnas 1, 2, ..., bb-1.
113
114         for k = bb : n+1
115
116         ele = ix(k);
117         if (k == bb)
118         jstart = 1;
119         else
120         jstart = bb;
121         end
122         for j = jstart : n+1
123         if (j < k)
124         imax = j-1;
125         else
126         imax = k-1;
127         end
128         if (imax < 1)
129         if (r(j) == n+1)
130         P(k,r(j)) = b(ele);
131         else
132         P(k,r(j)) = A(ele,r(j));
133         end
134         else
135         if (r(j) == n+1)
136         P(k,r(j)) = b(ele);
137         else
138         P(k,r(j)) = A(ele,r(j));
139         end
140         P(k,r(j)) = P(k,r(j)) - dot(P(k,r(1:imax)), P(1:imax,r(j
                )));
141         end
```

```
142         end
143         apiv = 0.0d0;
144         for j = k : n+1
145         t = P(k,r(j));
146         if (apiv < abs(t))
147         apiv = abs(t); pivot = t; jpivot = j;
148         end
149         end
150         if (apiv < eps)
151         refset = ix;
152         error('Rango insuficiente. Matriz singular.');
153         end
154         if (k < n+1)
155         jj = r(jpivot); r(jpivot) = r(k); r(k) = jj;
156         P(k,r(k+1:n+1)) = P(k, r(k+1:n+1))/pivot;
157         end
158         end
159
160         % Solucion subsistema P' lambda = -e_{n+1}
161
162         for j = bb : n+1                % sustitucion progresiva
163         if (r(j) == n+1)
164         sv(j) = -1.0d0;
165         else
166         sv(j) = 0.0d0;
167         end
168         sv(j) = sv(j) - dot(sv(1:j-1), P(1:j-1, r(j)));
169         end
170         lambda(n+1) = sv(n+1)/P(n+1,r(n+1));
171         for j = n : -1 : 1          % sustitucion regresiva
172         lambda(j) = (sv(j)-dot(lambda(j+1:n+1),P(j+1:n+1,r(j))))
                /P(j,r(j));
173         end
174
175         % Calcular epsilon para minimax con la referencia actual
176
177         epsilon = 1.0d0/sum(abs(lambda));
178         if (epsilon > lasteps)
179         lasteps = epsilon;
180
181         % Solución del sistema Px = epsilon*sign(lambda)
182         xr = sign(lambda)*epsilon;
183
184         % Sustitucion progresiva
185         w(1) = xr(1)/P(1,r(1));
186         for i = 2 : n+1
187         w(i) = (xr(i) - dot(w(1:i-1), P(i,r(1:i-1))))/P(i,r(i));
188         end
189
```

```
190         % Sustitucion regresiva
191         x(r(n+1)) = w(n+1);
192         for i = n : -1 : 1
193         x(r(i)) = w(i) - dot(x(r(i+1:n+1)), P(i,r(i+1:n+1)));
194         end
195
196         % Para purificar la solución, se toma x(n+1) igual a -1
197         % Calculo de los residuos de x en las ecuaciones que no
                son las
198         % de la referencia
199         ref = -x(n+1);
200         x = x/ref; epsilon = epsilon/ref;
201         ref = -1;
202
203         for j = n+2 : m
204         irow = ix(j);
205         t = -b(irow) + dot(x(1:n), A(irow,1:n));
206         if (abs(t) > ref)
207         ref = abs(t); al = j; s = sign(t);
208         end
209         end
210         if (ref <= epsilon)
211         done = true;
212         else
213
214         % Determino el indice de la ecuacion entrante del
                subsistema
215
216         k = ix(al);   % indice de ecuacion entrante
217
218         % Para determinar la ecuacion saliente resuelvo
219         % P' mu = A(k,r(i));
220
221         if (r(1) == n+1) % sustitucion progresiva
222         w(1) = b(k);
223         else
224         w(1) = A(k,r(1));
225         end
226         for i = 2 : n+1
227         if (r(i) == n+1)
228         w(i) = b(k);
229         else
230         w(i) = A(k,r(i));
231         end
232         w(i) = (w(i) - dot(w(1:i-1), P(1:i-1,r(i))));
233         end
234         w(n+1) = w(n+1)/P(n+1,r(n+1)); % sustitucion regresiva
235         for i = n : -1 : 1    % sustitucion regresiva
```

```
236         w(i) = (w(i) - dot(w(i+1:n+1), P(i+1:n+1,r(i))))/P(i,r(i
                ));
237         end
238
239         %s es el signo del residuo de mayor valor absoluto
240         %Encontrar índice con máxima ratio w(k)/lambda(k) * s (
                si
241         %alguna componente de lambda es cero salir con mensaje
                de
242         %interrupcion por lambda  cero)
243
244         ref = lambda(n+1); bb = n+1;
245         if (abs(ref) < eps)         %test si lambda es cero
246         refset = ix;
247         error('Interrupcion: lambda cero')
248         end
249         ref = w(n+1)/ref*s;
250         for j = 1 : n
251         t = lambda(j);
252         if (abs(t) < eps)      %test si lambda es cero
253         refset = ix;
254         error('Interrupcion: lambda cero')
255         end
256         t = w(j)/t*s;
257         if (t > ref)
258         bb = j; ref = t;
259         end
260         end
261         ix(al) = ix(bb); ix(bb) = k; a1 = 1;
262         done = false;
263         end
264         else          %restaura la referencia anterior, y
                procede a recomputar
265         epsilon = lasteps; a1 = 2;
266         jj = ix(al); ix(al) = ix(bb); ix(bb) = jj;
267         ref = -1;
268         for j = n+2 : m
269         ii = ix(j);
270         t = -b(ii) + dot(x(1:n), A(ii,1:n));
271         if (abs(t) > ref)
272         ref = abs(t);
273         end
274         end
275         end
276         iter = iter+1;
277         end
278         refset = ix;
279         end
```

2.3.4. Ejemplos

2.3.4.1. Ejemplo 1

A modo de ilustración del algoritmo, consideramos como primer test la solución minimax del sistema lineal

$$\begin{bmatrix} 11 & -8 & -6 \\ 0 & -15 & -12 \\ -13 & -3 & 10 \\ 7 & 8 & 2 \\ 10 & -7 & 9 \\ 0 & -5 & 5 \\ 7 & 10 & 9 \\ -15 & 0 & 15 \\ -15 & 3 & -15 \\ 2 & 5 & 14 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -68 \\ -54 \\ 11 \\ 3 \\ -64 \\ -19 \\ 13 \\ 30 \\ 72 \\ -5 \end{bmatrix}$$

tomado de [7].

La solución es $\mathbf{x} = [-3, 4, -1]^T$, con un residuo cuya norma infinito viene dada por $\|\mathbf{r}\|_\infty = 6{,}0$. La solución se alcanza en tres iteraciones, con la referencia final (componentes del residuo donde se alcanza la norma infinito) formada por las ecuaciones que corresponden a las filas 2,3,4 y 6 de la matriz A.

2.3.4.2. Ejemplo 2

Para considerar un problema más exigente tomamos como matriz A una matriz aleatoria, de tamaño 100×30, con sus elementos distribuidos de forma uniforme en el intervalo $[0, 1]$. Para definir $\mathbf{b}$, determinamos un vector $\mathbf{x}$ aleatorio, con componentes distribuidas de forma uniforme en $[0, 1]$, y definimos

$$\mathbf{b} = A * \mathbf{x} + \texttt{tol} * \texttt{rand}(\texttt{100}, \texttt{1}).$$

En un ejemplo de este tipo de problema (con `tol` = 0,01), la solución se alcanzó en 60 iteraciones, con una solución cuyos residuos máximos sólo se diferenciaban en unas pocas unidades en la última cifra significativa de la aritmética coma flotante de Matlab.

Este mismo ejemplo se ha llevado a cabo 50 veces gracias a la siguiente función auxiliar.

```
function [itertotal, epsxtotal] = ejemplo (n)
2       for i = 1:n
3       A = rand(100,30);
```

```
4         x = rand(30,1);
5         b = A*x + 0.01*rand(100,1);
6         [x, refset, epsilon, iter] = alg_bargol( A, b );
7         epsxtotal(i) = epsilon;
8         itertotal(i) = iter;
9         end
end
```

Debido a esta función hemos hallado el promedio de las iteraciones totales este es: 70 y la desviación promedio minimax 0,003829844133061.

Realizando este mismo ejemplo comparando el algoritmo ascendente y el algoritmo ascendente con descomposición LU, se obtiene que el primero lo hace en una media de 73 iteraciones, mientras que el segundo alcanza la solución minimax en 70 operaciones. De esta manera llegamos a la conclusión de que el algoritmo con descomposición LU es más eficiente.

2.4. Algoritmo del descenso

En esta sección se considera el método general descendente. Se debe minimizar la siguiente función:

$$\delta(\mathbf{x}) = \max_{1 \leq i \leq m} \left\{ \left\langle \mathbf{A^i}, \mathbf{x} \right\rangle - b_i \right\}.$$

Al igual que en el algoritmo ascendente, primero se va a describir el algoritmo para un sistema de ecuaciones con una única incógnita. Empezando con un punto x_0 cualquiera, se define $M = \{i : r_i(x_0) = \delta(x_0)\}$. Si M contiene dos índices $a_j a_k < 0$, entonces x_0 es la solución. Al no importar si $\mathbf{x}$ aumenta o disminuye, uno de los $r_j(\mathbf{x})$ y $r_k(\mathbf{x})$ se incrementará, esto hace que $\delta(\mathbf{x})$ aumente. Por otro lado, un mínimo local es necesariamente un mínimo global al ser una función convexa. Si x_0 no es la solución, se selecciona $j \in M$ para el que $|a_j|$ es mínimo. Para que $\delta(\mathbf{x})$ decrezca, es necesario y suficiente que $r_j(\mathbf{x})$ decrezca. La dirección en que esto se puede conseguir es hacia la derecha si $a_j < 0$ y hacia la izquierda si $a_j > 0$. Se selecciona j de tal manera que se asegure que $r_j(\mathbf{x})$ y $\delta(\mathbf{x})$ son idénticas a lo largo del segmento que une x_0 con el siguiente punto. Este se encontrará entre los puntos y_i definido por las ecuaciones $r_j(y_i) = r_i(y_i)$, y es el primero que se encuentra a derecha o izquierda de x_0, dependiendo del signo de a_j.

La idea general del método es proceder de manera descendente de vértice a vértice en la hipersuperficie en un espacio de dimensión $n+1$ cuya ecuación es

$$z = \delta(\mathbf{x})$$

Sea $\mathbf{x_0}$ cualquier vector inicial. Renumerando los residuos según convenga, se asume:

$$\delta\left(\mathbf{x^0}\right) = r_1\left(\mathbf{x^0}\right) = r_2\left(\mathbf{x^0}\right) = \ldots = r_k\left(\mathbf{x^0}\right) > r_{k+i}\left(\mathbf{x^0}\right) \qquad (i \geq 1)$$

Se busca la dirección en la que se debe mover desde $\mathbf{x^0}$ en la cual los residuos $r_1, \ldots, r_k$ disminuyan a la misma velocidad. Se sigue en esta dirección hasta que una función residual $(k+1)$-ésima alcance el primer k. Reaplicando esta técnica se llega en un máximo de n pasos a un vértice, donde hay $n+1$ residuos máximos iguales. Ahora, la velocidad de cambio en r_i mientras se mueve desde $\mathbf{x^0}$ en dirección $\mathbf{y}$ es $\left\langle \mathbf{A^i}, \mathbf{y} \right\rangle$. De hecho,

$$\frac{d}{d\lambda} r_i\left(\mathbf{x^0} + \lambda \mathbf{y}\right) = \frac{d}{d\lambda}\left[\left\langle \mathbf{A^i}, \mathbf{x^0} + \lambda \mathbf{y} \right\rangle - b_i\right] = \left\langle \mathbf{A^i}, \mathbf{y} \right\rangle$$

Por lo que una dirección $\mathbf{y}$ en la que los residuos $r_1, \ldots, r_k$ permanecen iguales decreciendo a una velocidad común puede determinarse resolviendo las ecuaciones:

$$\left\langle \mathbf{A^i}, \mathbf{y} \right\rangle = -1 \qquad (i = 1, \ldots, k) \tag{2.14}$$

Asumiendo que cada conjunto de n vectores $\mathbf{A^i}$ es independiente, es decir, que se cumple la condición de Haar, entonces esta condición sobre $\mathbf{y}$ se cumple fácilmente mientras $k \leq n$. De hecho, se puede tomar $\mathbf{y}$ para que sea de la forma:

$$\mathbf{y} = \sum_{\mathbf{j=1}}^{\mathbf{k}} \mathbf{c_j A^j}$$

las ecuaciones entonces serán

$$\sum_{j=1}^{k} c_j \left\langle \mathbf{A^i}, \mathbf{A^j} \right\rangle = -1 \quad (i = 1, \ldots, k)$$

Es claro que la matriz de Gram no es singular. Habiendo determinado el vector $\mathbf{y}$ con la propiedad (2.14), el próximo punto debe ser de la forma $\mathbf{x^1} = \mathbf{x^0} + \lambda \mathbf{y}$ donde λ es el coeficiente más pequeño positivo en el que se tienen k + 1 residuos máximos iguales. Asumimos ahora que se conoce un punto $\mathbf{x^0}$ donde se producen n + 1 residuos máximos iguales,

$$\delta\left(\mathbf{x^0}\right) = r_1\left(\mathbf{x^0}\right) = \ldots = r_{n+1}\left(\mathbf{x^0}\right) > r_{n+i}\left(\mathbf{x^0}\right) \qquad (i > 1)$$

Puede ocurrir que $\mathbf{x^0}$ sea solución. Esto será el caso si y solo si el sistema de desigualdades lineales:

$$\left\langle \mathbf{A^i}, \mathbf{y} \right\rangle < 0 \quad (i = 1, \ldots, n+1)$$

es inconsistente. Si este sistema es consistente, entonces por el teorema de desigualdades lineales (teorema 1.1.5) no se encuentra en la envolvente convexa de $\{\mathbf{A^1}, \dots \mathbf{A^{n+1}}\}$. En consecuencia, si se resuelven las siguientes ecuaciones para $\theta_1, \dots, \theta_{n+1}$,

$$\sum_{i=1}^{n+1} \theta_i \mathbf{A^i} = \mathbf{0} \qquad \sum_{\mathbf{i=1}}^{\mathbf{n+1}} \theta_{\mathbf{i}} = \mathbf{1}$$

entonces al menos un coeficiente θ_i será negativo. Ahora, saliendo desde el vértice asociado con el punto $\mathbf{x^0}$ hay ciertos "bordes" que son variedades lineales unidimensionales, a lo largo de las cuales, n residuos permanecen iguales entre sí. No todos estos bordes están realmente en la superficie $z = \delta(\mathbf{x})$. Para cada conjunto de n índices seleccionados desde $\{1, \dots, n+1\}$, hay un borde a lo largo del cual estos n residuos son iguales. El residuo restante, digamos r_j, cambiará a un ritmo diferente. La dirección de este borde será la obtenida resolviendo un sistema como el siguiente para el vector $\mathbf{y^j}$:

$$\left\langle \mathbf{A^i}, \mathbf{y^j} \right\rangle = -1 \qquad (i = 1, \dots, n+1;\ i \neq j)$$

En la dirección y^j, el residuo r_j cambiará a un ritmo $\left\langle \mathbf{A^j}, \mathbf{y^j} \right\rangle$, y esto puede ser mayor o menor que -1. Si $\left\langle \mathbf{A^j}, \mathbf{y^j} \right\rangle < -1$, entonces el borde correspondiente permanece en la superficie. Para calcular $\left\langle \mathbf{A^j}, \mathbf{y^j} \right\rangle$:

$$0 = \left\langle 0, y^j \right\rangle = \sum_{i=1}^{n+1} \theta_i \left\langle \mathbf{A^i}, \mathbf{y^j} \right\rangle = \sum_{\substack{i=1 \\ i \neq j}}^{n+1} -\theta_i + \theta_j \left\langle \mathbf{A^i}, \mathbf{y^j} \right\rangle$$

Por lo tanto

$$\left\langle \mathbf{A^j}, \mathbf{y^j} \right\rangle = \frac{1}{\theta_j} \sum_{\substack{i=1 \\ i \neq j}}^{n+1} \theta_i = \frac{1 - \theta_j}{\theta_j} = \frac{1}{\theta_j} - 1$$

Uno de estos ritmos será menor que -1, ya que al menos uno de los θ_j es negativo. Si j es un índice, el siguiente punto es de la forma $\mathbf{x^0} + \lambda \mathbf{y^j}$, donde tomamos λ para que sea el coeficiente más pequeño donde hay $n+1$ residuos máximos iguales. Desde aquí para seguir calculando se podrá utilizar el teorema 2.3.3, puesto que debemos de volver a calcular C únicamente modificando la fila que hemos cambiado.

2.4.1. Código en MATLAB

```
1   % Algoritmo descendente
2
3   function [x, iter] = alg_descenso(A,b,x)
4
5   A=[A;-A];
6   b=[b;-b];
7   [m,n] = size(A);
8   resto = zeros(m,1);
9
10  % seleccionamos M, es el conjunto de índices que tiene
        resto = delta
11  resto = A*x - b;
12  [delta, M] = max(resto);
13  M = find(resto==delta);
14  k = length(M);
15
16  MM = setdiff([1:m],M); % conjunto de filas donde no se
        alcanza el delta
17  iter = 1;
18
19  while (k - 1) < n
20
21  kones = - ones(k,1);
22  Gram = A(M,:) * A(M, :)';
23  coeff = kones'/Gram;
24  y = coeff*A(M,:);
25
26  lambda = (delta - resto(MM))./( A(MM,:)*y' + 1);
27  posit = find(lambda > 0); %me encuentra las posiciones
        de los positivos
28  [ratlam, pos] = min(lambda(posit));
29
30  J = MM(posit(pos)); % tengo que coger el índice de MM
        que tenga lambda mínimo
31  M = union(M,J);
32  x = x + ratlam*y';
33  resto = A*x - b;
34  k = length(M);
35  % hay que quitar el índice que hemos cogido de J
36  MM = setdiff([1:m], M);
37  delta = delta - ratlam;
38  iter = iter+1;
39  end
40
41  resto = A*x-b;  % vector de residuos con al menos n+1
        residuos iguales a delta(x)
42
43  % caso de que SI
```

```
44        C = inv([ones(k,1), A(M,:)]);
45
46        while (min(C(1,:)) < 0.0)
47        %p donde C(1,:) sea mínimo
48        [cmin, p] = min(C(1,:));
49
50        %En la fila donde se encuentre el minimo C_p^0, dividir
              cada
51        %elemento de la columna por el minimo
52        y = C(2:end,p)/cmin;
53
54        t = (delta - resto(MM))./(1 + A(MM,:)*y);
55
56        %busco el indice del ratio positivo más pequeño
57        posit = find(t>0);    %me encuentra las posiciones de
              los positivos
58        if (isempty(posit))
59        error('no existe t positivo');
60        end
61        [tmin, alpha] = min(t(posit));
62
63        x = x + tmin*y;
64        AA = [ones(1), A(MM(posit(alpha)),:)];
65        lambda = AA * C;
66
67        %seleccionamos beta para que sea maximo
68        [bmax,beta] = max(lambda./C(1,:));
69
70        %mismo que algoritmo ascendente.
71        C(:,beta) = C(:,beta)/lambda(beta);
72
73        for i = 1 : n+1
74        if i ~= beta
75        C(:,i) = C(:,i)-lambda(i)*C(:,beta);
76        end
77        end
78
79        M(beta) = MM(posit(alpha));
80        MM = setdiff([1:m], M);
81        iter = iter+1;
82        resto = A*x - b;
83        delta = delta - tmin;
84        end
85        disp(x);
86        disp(delta);
87        disp(resto);
88        disp(M);
89        end
```

2.4.1.1. Problemas test

2.4.1.2. Ejemplo 1

Consideramos de nuevo el problema test de [7]

$$\begin{bmatrix} 11 & -8 & -6 \\ 0 & -15 & -12 \\ -13 & -3 & 10 \\ 7 & 8 & 2 \\ 10 & -7 & 9 \\ 0 & -5 & 5 \\ 7 & 10 & 9 \\ -15 & 0 & 15 \\ -15 & 3 & -15 \\ 2 & 5 & 14 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -68 \\ -54 \\ 11 \\ 3 \\ -64 \\ -19 \\ 13 \\ 30 \\ 72 \\ -5 \end{bmatrix}$$

Con una aproximación inicial igual a $\mathbf{x}_0 = [1, -1, 1]^T$, se obtiene la solución minimax $[-3, 4, -1]^T$ en 15 iteraciones. El número de iteraciones de descenso (cada iteración requiere la solución de dos sistemas lineales) es con diferencia bastante mayor que el número de iteraciones requeridas por el algoritmo de ascenso. Una razón posible es la falta de optimalidad en la elección de las direcciones de descenso que toma el algoritmo.

Capítulo 3

Aproximación Chebyshev por polinomios y otras familias de funciones lineales

En este capítulo se considerará el problema de aproximar una función continua f definida en un intervalo $[a, b]$ por un polinomio de grado menor o igual que n, $P \in \mathscr{P}_n$. Se desea minimizar expresiones con la siguiente forma:

$$\max_{a \le x \le b} |f(x) - P(x)| \tag{3.1}$$

o

$$\max_{0 \le i \le m} |f(x_i) - P(x_i)|, \tag{3.2}$$

donde $X = \{x_0, \ldots, x_m\}$ es un subconjunto finito de abscisas prefijadas en $[a, b]$, distintas dos a dos.

Se hablará de un problema de aproximación más general en el que los monomios $1, x, x^2, \ldots, x^n$ serán reemplazados por otras familias de funciones continuas $g_0, g_1, \ldots, g_n$. Nos referiremos a funciones de la forma $\sum_{i=0}^{n} c_i g_i$ con el nombre de polinomios generalizados de grado n. Notemos que, en particular, cuando en la expresión (3.2) el número de abscisas se toma igual a $n+1$ (la dimensión del espacio $\mathscr{P}_n$ o el espacio de polinomios generalizados), el problema se reformula como uno de interpolación de Lagrange.

Si X es solamente un conjunto finito de puntos, entonces el problema se reduce a encontrar los coeficientes del aproximante

$$P_n(x) = c_0 + c_1 x + \cdots + c_n x^n$$

mediante la solución minimax del sistema sobredeterminado de ecuaciones lineales

$$V^T \mathbf{c} = \mathbf{b}$$

donde $\mathbf{c} = [c_0, \ldots, c_n]^T$, $\mathbf{b} = [f(x_0), \ldots, f(x_n)]^T$ y V es la matriz de Vandermonde asociada a las abscisas $x_0, \ldots, x_m$.

3.1. Familias lineares generales

Pese a que el algoritmo que vamos a desarrollar va a ser para polinomios, vamos a exponer en esta sección distintos teoremas que se cumplen para cualquier subespacio generado por una familia finita de funciones. Se debe generalizar el concepto de polinomio para incluir combinaciones lineales de otras funciones continuas como, $g_1, g_2, \ldots, g_n$.

Tales funciones son continuas en un espacio métrico compacto fijado X (que generalmente será un intervalo $[a, b]$) y, como ya se ha dicho, sus combinaciones lineales $\sum_{i=1}^{n} c_i g_i$ se llaman polinomios generalizados. Con el siguiente teorema se garantizará que para cada $f \in C[X]$ existe al menos un polinomio generalizado de mejor aproximación a f.

Teorema 3.1.1 (Teorema de la existencia)**.** *Un subespacio vectorial de dimensión finita de un espacio vectorial normado contiene, al menos, un punto con distancia mínima desde un punto fijado.*

Demostración. Sea M dicho subespacio y g el punto fijado. Sea f_0 un punto arbitrario de M. Entonces el punto buscado se encuentra en el conjunto:

$$\{f : f \in M, \|f - g\| \leq \|f_0 - g\|\}.$$

Por ser un subconjunto cerrado y acotado de un subespacio de dimensión finita es compacto. Por lo tanto, contiene un punto de distancia mínima desde g. □

Teorema 3.1.2 (Teorema de caracterización)**.** *Para que los coeficientes que denotamos $c_1, \ldots, c_n$ hagan que la norma uniforme del error $r := \Sigma c_i g_i - f$ sea mínima, es necesario y suficiente que el origen del espacio $\mathbb{R}^n$ permanezca en la envolvente convexa del conjunto de puntos $\{r(x)\hat{\mathbf{x}} : |r(x)| = \|r\|_\infty\}$, donde $\hat{\mathbf{x}}$ denota la n-upla $\{g_1(x), \ldots, g_n(x)\}$.*

Demostración. Se va a demostrar haciendo uso de la reducción al absurdo. Se supone que $\|r\|_\infty$ no es mínimo. Entonces, existe algún vector $\mathbf{d} = [d_1, \ldots, d_n]^T$, que cumple $\| \sum (c_i - d_i) g_i - f \|_\infty < \| \sum c_i g_i - f \|_\infty$. Ahora, sea $X_0 = \{x \in X : |r(x)| = \|r\|_\infty\}$. Para $x \in X_0$ se tiene:

$$\begin{aligned} [r(x) - \sum d_i g_i(x)]^2 &< [r(x)]^2 \\ \left[r^2(x) - 2r(x) \sum d_i g_i(x) + \left(\sum d_i g_i(x)\right)^2\right] &< [r(x)]^2 \end{aligned}$$

de donde se obtiene el siguiente sistema de desigualdades lineales, que debe satisfacerse:

$$r(x)\sum d_i g_i(x) \equiv r(x)\,\langle \mathbf{d}, \hat{\mathbf{x}}\rangle > 0 \qquad (x \in X_0). \tag{3.3}$$

Por el teorema de las desigualdades lineales (teorema 1.1.5) se sabe que 0 permanece fuera de la envolvente convexa de $\{r(x)\hat{x} : x \in X_0\}$.

Para ver la otra implicación, se supone que 0 permanece fuera de la envolvente convexa de $\{r(x)\hat{\mathbf{x}} : x \in X_0\}$. Por el teorema de las desigualdades lineales, sabemos que existe un vector $\mathbf{d}$ tal que la desigualdad (3.3) es cierta para $x \in X_0$. Como X_0 es compacto, el número $\epsilon = \text{mín}_{x\in X_0}\, r(x)\,\langle \mathbf{d}, \hat{\mathbf{x}}\rangle$ es positivo. Sea $X_1 = \{x \in X : r(x)\,\langle \mathbf{d}, \hat{\mathbf{x}}\rangle \leq \epsilon/2\}$. X_1 es un conjunto cerrado que no contiene puntos de X_0. Esto se da debido a que $|r(x)|$ alcanza su extremo superior E en X_1 y $E < \|r\|_\infty$. Se prueba que para algún $\lambda > 0$, $\|r - \lambda \sum d_i g_i\|_\infty < \|r\|_\infty$.

Se toma $x \in X_1$ suponiendo que $0 < \lambda < (\|r\|_\infty - E)/\|\sum d_i g_i\|$. Entonces,

$$|r(x) - \lambda \sum d_i g_i(x)| \leq |r(x)| + \lambda|\sum d_i g_i(x)| \leq E + \lambda\|\sum d_i g_i\| < \|r\|$$

Si $x \notin X_1$ y se asume que $0 < \lambda < \epsilon/\|\sum d_i g_i\|^2$. Entonces

$$\begin{aligned}\left[r(x) - \lambda \sum d_i g_i(x)\right]^2 &= [r(x)]^2 - 2\lambda r(x)\,\langle \mathbf{d}, \hat{\mathbf{x}}\rangle + \lambda^2 \left[\sum d_i g_i(x)\right]^2 < \\ &< \|r\|^2 + \lambda(-\epsilon + \lambda\|\sum d_i g_i\|^2) < \|r\|^2\end{aligned}$$

□

Este teorema es la versión continua del teorema 2.1.2. Para algunos tipos de polinomios generalizados, la caracterización de la mejor aproximación puede estar dada de una manera mucho más conveniente y, para ella, será necesaria la propiedad de la condición de Haar.

Definición 3.1.3. Se dice que un sistema de funciones $\{g_1, \ldots, g_n\}$ satisface la condición de Haar, si cada g_i es continua y si cada conjunto de n vectores de la forma $\hat{\mathbf{x}} = [g_1(x), \ldots, g_n(x)]$ es independiente. Se puede expresar también como que cada determinante:

$$D\,[x_1, \ldots, x_n] = \begin{vmatrix} g_1(x_1) & \cdots & g_n(x_1) \\ \vdots & \ldots & \vdots \\ g_1(x_n) & \cdots & g_n(x_n) \end{vmatrix}$$

asociado a n puntos distintos $x_1, \ldots, x_n$ del dominio de definición de las g_i, es distinto de cero.

Nótese que, cuando las funciones $g_i, i = 1, \ldots, n$ son las funciones base $\{1, x, x^2, \ldots, x^n\}$ de $\mathscr{P}_n$, entonces la condición de Haar se satisface automáticamente en cualquier intervalo y para cualquier n. Ello es consecuencia de que entonces el determinante $D[x_0, \ldots, x_n]$ no es otro que el determinante de la matriz de Vandermonde asociada a las abscisas $x_0, \ldots, x_n$.

Nótese que $\{g_1, \ldots, g_n\}$ satisface la condición de Haar si 0 es la única función de la forma $\sum_{i=1}^{n} c_i g_i$ que tiene n o más ceros en el intervalo $[a, b]$. Los siguientes teoremas se cumplen para cualquier familia lineal generalizada $\langle g_1, \ldots, g_n \rangle$ y simplemente los enunciaremos, dejando su demostración para cuando nos centremos en el espacio de polinomios de grado máximo n, $\mathscr{P}_n$.

Lema 3.1.4. *Sea $\{g_1, \ldots, g_n\}$ un sistema de elementos de $C[a, b]$ que satisface la condición de Haar. Sea $a \leq x_0 < \ldots < x_n \leq b$, y sean $\lambda_0, \ldots, \lambda_n$ constantes distintas de cero. Para que 0 permanezca en la envolvente convexa de las n-uplas $\lambda_0 \hat{x}_0, \ldots, \lambda_n \hat{x}_n$ es necesario y suficiente que los λ vayan alternándose en el signo, es decir, $\lambda_i \lambda_{i-1} < 0$ para $i = 1, \ldots, n$.*

Teorema 3.1.5 (Teorema de alternancia)**.** *Sea $\{g_1, \ldots, g_n\}$ un conjunto de elementos de $C[a, b]$ que satisface la condición de Haar, y sea X cualquier subconjunto cerrado de $[a, b]$. Para que un polinomio generalizado $P = \sum c_i g_i$ sea la mejor aproximación en X de una función dada $f \in C[X]$, es necesario y suficiente que la función de error $r = f - P$ presente en X, al menos, $n+1$ cambios de signo: $r(x_i) = -r_(x_{i-1}) = \pm\|\mathbf{r}\|_\infty$, con $x_0 < \ldots < x_n$ y $x_i \in X$. Se denota $\|\mathbf{r}\|_\infty = \max_{x \in X} |r(x)|$.*

Si se conociera la localización de $n + 1$ puntos extremos de la curva de error, es fácil obtener la aproximación polinomial (generalizada) óptima resolviendo el sistema de ecuaciones lineales

$$f(x_i) - \sum_{k=1}^{n} c_k g_k(x_i) = \pm\|\mathbf{r}\|_\infty.$$

Sin embargo, la localización de estos extremos la mayor parte de las veces requiere de la solución de sistemas de ecuaciones no linales.

Como aplicación del teorema de alternancia se puede probar un teorema de no existencia. Antes definimos lo que se entiende por un sistema de Markoff de funciones generalizadas.

Definición 3.1.6. Un sistema ordenado finito o infinito de funciones continuas $\{g_1, g_2, \ldots\}$, en un intervalo $[a, b]$, se denomina sistema de Markoff si cada segmento inicial $\{g_1, \ldots, g_n\}$ satisface la condición de Haar.

Por ejemplo, los monomios $1, x, x^2, \ldots$ forman un sistema de Markoff en cualquier intervalo.

Teorema 3.1.7. *Sea $\{g_1, g_2, \ldots\}$ un sistema infinito de Markoff en $[a, b]$. Sea M el subespacio vectorial cerrado de $C[a, b]$ generado por los elementos g_i. Entonces, ningún punto de $C[a, b]$ que no pertenezca a M tiene una mejor aproximación en M.*

Demostración. Razonamos mediante reducción al absurdo. Sea $f \notin M$ y supongamos que existe un punto g perteneciente al subespacio vectorial M que es el más cercano a f, y supongamos $\|f - g\| = \epsilon > 0$. Entonces, 0 es el punto más cercano de M a $f - g$. Sea M_n el subespacio de dimensión finita generado por $\{g_1, \ldots, g_n\}$. Entonces, 0 es el punto de M_n más cercano a $f-g$. Por el teorema de alternancia, $f - g$ debe tener al menos $n + 1$ oscilaciones de amplitud ϵ. Debido a que esto es cierto para todo n, $f - g$ no puede ser continua. □

Otra aplicación del teorema de alternancia puede ser tomando X como cualquier conjunto de $n + 2$ puntos de la línea real. Se tiene entonces el siguiente caso:

Teorema 3.1.8. *Sean polinomios cualesquiera P y Q de grado $\leq n + 1$ que cumplen las siguientes condiciones: $P(x_i) = f(x_i)$ y $Q(x_i) = (-1)^i$, donde $x_0 < \ldots < x_{n+1}$. Entonces, la polinomial de grado $\leq n$ que mejor aproxima f en $\{x_i\}$ es $P - \lambda Q$, donde λ es elegido para que la polinomial resultante sea de grado $\leq n$. El error de aproximación es $|\lambda|$.*

En el siguiente teorema se tiene un sistema de funciones continuas que denotamos $\{g_1, \ldots, g_n\}$ satisfaciendo la condición de Haar. Se denota por $E(f)$ el ínfimo de $\|P-f\|$ como P varía en todos los polinomios generalizados, $P = \sum c_i g_i$.

Teorema 3.1.9 (Teorema de La Vallée Poussin)**.** *Si P es un polinomio generalizado tal que $f - P$ asume valores alternados positivos y negativos en $n + 1$ puntos consecutivos x_i de $[a, b]$, entonces $E(f) \geq \text{mín}_i\, |f(x_i) - P(x_i)|$.*

Demostración. Si la conclusión fuera falsa, existiría un polinomio generalizado P_0 tal que $\|f - P_0\| < \text{mín}_i\, |f(x_i) - P(x_i)|$. Entonces, el polinomio generalizado $P_0 - P = (f - P) - (f - P_0)$ es alternativamente positivo y negativo en los puntos x_i y consecuentemente, desaparece en n puntos. Pero esto no es posible. □

3.2. La teoría de la aproximación minimax

Hemos visto ya que la aproximación minimax en un conjunto $\mathscr{A}$ a una función f en $\mathscr{C}[a,b]$ es el elemento de $\mathscr{A}$ que minimiza la siguiente expresión:

$$\|f-p\|_\infty = \max_{a\le x\le b} |f(x)-p(x)|, \quad p\in\mathscr{A}. \tag{3.4}$$

En esta sección se verán las condiciones que se deben satisfacer para que haya una mejor aproximación cuando $\mathscr{A}$ es un espacio vectorial de dimensión finita. Pese a que existen teorías en las que $\mathscr{A}$ puede ser cualquier espacio vectorial, en este apartado nos vamos a centrar en el espacio $\mathscr{P}_n$, es decir, el espacio de polinomios de grado máximo n. Sin embargo, estos resultados se generalizan a todo subespacio $\mathcal{A}$ de dimensión finita que cumpla la condición de Haar, una condición que generaliza la introducida en el capítulo anterior y que introduciremos más adelante

Veremos además una serie de propiedades que se cumplen cuando se tiene la condición de Haar, incluyendo el resultado importante de que la mejor aproximación en la norma uniforme es única. La condición de Haar, nos garantizará también un método, denominado el algoritmo de intercambio, para poder calcular la mejor aproximación.

Supongamos que se tiene una aproximación inicial p^* de $\mathscr{A}$ a f. Si quisiéramos reducir el error máximo de la aproximación inicial, deberíamos poder encontrar un elemento p en $\mathscr{A}$, no nulo, tal que la desigualdad

$$\|f-(p^*+\theta p)\|_\infty < \|f-p^*\|_\infty \tag{3.5}$$

se satisfaciera para algún valor del parámetro θ.

3.2.1. Reducción del error en una aproximación de prueba

Sea p^* una aproximación en $\mathscr{A}$ a una función $f\in\mathscr{C}[a,b]$. Queremos mejorar la aproximación determinando $p\in\mathcal{A}$ y un número real θ, tal que se satisfaga la condición (3.5). Denotemos con $\mathscr{Z}_M$ el conjunto de puntos en los que la función error, definida como

$$e^*(x) = f(x)-p^*(x), \quad a\le x\le b \tag{3.6}$$

alcanza sus extremos. Este conjunto está caracterizado por la condición

$$|e^*(x)| = \|e^*\|_\infty, \quad \forall x\in\mathscr{Z}_M.$$

Supongamos que p^* no es óptimo. Sea $(p^* + \theta p)$ la mejor aproximación. Entonces, los puntos en $\mathscr{Z}_M$ deben satisfacer la desigualdad

$$|e^*(x) - \theta p(x)| < |e^*(x)|, \quad x \in \mathscr{Z}_M.$$

Supongamos, sin pérdida de generalidad, que θ es positivo. Por la desigualdad anterior, si x pertenece a $\mathscr{Z}_M$, entonces necesariamente $e^*(x)$ tiene el mismo signo que $p(x)$. Se deriva entonces que p^* es la aproximación minimax en $\mathscr{A}$ a f, si no existe una función p en $\mathscr{A}$ que satisfaga la condición

$$[f(x) - p^*(x)]p(x) > 0, \quad x \in \mathscr{Z}_M. \tag{3.7}$$

El resto de la sección establece el recíproco de este resultado: es decir, si se verifica la desigualdad (3.7) para algún $p \in \mathcal{A}$, entonces existe un valor positivo de θ, tal que produce la reducción (3.5).

Generalizamos el problema de minimizar $\|f - p\|_\infty$ en un intervalo $[a, b]$ al problema de minimizar la expresión

$$\max_{x \in \mathscr{Z}} |f(x) - p(x)|, \quad p \in \mathscr{A} \tag{3.8}$$

donde $\mathscr{Z}$ es cualquier subconjunto cerrado de $[a, b]$. Este conjunto $\mathscr{Z}$ puede ser el propio $[a, b]$, pero también puede ser un subconjunto finito de $[a, b]$. Esta última circunstancia será la característica de los problemas minimax en cada iteración del algoritmo del intercambio. Por tanto, los teoremas que siguen se aplican a ambas situaciones.

Teorema 3.2.1. *Sea $\mathscr{A}$ un subespacio vectorial de $\mathscr{C}[a, b]$, sea f una función cualquiera en $\mathscr{C}[a, b]$, sea $\mathscr{Z}$ cualquier subconjunto cerrado de $[a, b]$, sea p^* cualquier elemento de $\mathscr{A}$, y sea $\mathscr{Z}_M$ el subconjunto de puntos de $\mathscr{Z}$ en el cual el error $\{|f(x) - p^*(x)| : x \in \mathscr{Z}\}$ toma sus valores máximos. Entonces, p^* es un elemento de $\mathscr{A}$ que minimiza $\max_{x \in \mathscr{Z}} |f(x) - p(x)|$ si y solo si no existe una función p en $\mathscr{A}$ que satisfaga la condición $[f(x) - p^*(x)]p(x) > 0$, para todo $x \in \mathcal{Z}_M$.*

Demostración. La primera implicación está clara por como se ha ido desarrollando en los párrafos previos a este teorema, pero la prueba se extiende sin dificultad a cuando $[a, b]$ se reemplaza por $\mathscr{Z}$, un subconjunto cerrado del intervalo $[a, b]$.
Por lo tanto, falta demostrar que, si se tiene la condición $[f(x) - p^*(x)]p(x) > 0$ para todo $c \in \mathcal{Z}_M$, entonces podemos reducir el error de modo que la desigualdad

$$\max_{x \in \mathscr{Z}} |e^*(x) - \theta p(x)| < \max_{x \in \mathscr{Z}} |e^*(x)|$$

se satisface para algunos valores de θ, donde e^* es la función de error descrita anteriormente.

Sea θ positivo y se asume, sin pérdida de generalidad, $|p(x)| \leq 1$ con $a \leq x \leq b$. Debemos tener un cuidado especial con las abscisas x para las cuales los signos de $e^*(x)$ y $p(x)$ son opuestos. Por lo tanto, definimos el conjunto $\mathscr{Z}_0$ formado por los elementos $x \in \mathcal{Z}$ que satisfacen la condición $p(x)e^*(x) \leq 0$. Como este conjunto es cerrado y, como $\mathscr{Z}_0$ y $\mathscr{Z}_M$ no tienen puntos en común, el número

$$d = \max_{x \in \mathscr{Z}_0} |e^*(x)|$$

satisface $d < \max_{x \in \mathscr{Z}} |e^*(x)|$. Si $\mathscr{Z}_0$ fuera vacío, pondríamos d igual a cero. La desigualdad buscada se obtiene cuando θ toma el valor positivo $\theta = \frac{1}{2}[\max_{x \in \mathscr{Z}} |e^*(x)| - d]$. Como el conjunto $\mathscr{Z}$ es cerrado, sea ξ un elemento de $\mathscr{Z}$ que satisfaga la ecuación:

$$|e^*(\xi) - \theta p(\xi)| = \max_{x \in \mathscr{Z}} |e^*(x) - \theta p(x)|$$

Si ξ pertenece a $\mathscr{Z}_0$, se obtiene

$$\max_{x \in \mathscr{Z}} |e^*(x) - \theta p(x)| = |e^*(\xi)| + |\theta p(\xi)| \leq d + \theta$$

En esta última expresión, el último término depende de los límites de d mencionados previamente sin olvidar que $|p(x)| \leq 1$. Por lo tanto, la desigualdad buscada se consigue gracias a la desigualdad de $d < \max_{x \in \mathscr{Z}} |e^*(x)|$ y la ecuación que definía a θ.

Alternativamente, si ξ no pertenece a $\mathscr{Z}_0$, los signos de los términos $e^*(\xi)$ y $p(\xi)$ son los mismos, lo que lleva a la siguiente desigualdad estricta:

$$|e^*(\xi) - \theta p(\xi)| < \max\left[|e^*(\xi)|, |\theta p(\xi)|\right],$$

que prueba la desigualdad buscada. □

Este teorema muestra que, para ver si una aproximación es óptima, solo se necesitan considerar los valores extremos de la función de error. Específicamente, hay que ver si la condición

$$\left[f(x) - p^*(x)\right] p(x) > 0, \quad x \in \mathscr{Z}_M \tag{3.9}$$

se cumple para alguna función p en $\mathscr{A}$.

3.3. Teorema de caracterización y la condición de Haar

Si el conjunto $\mathscr{A}$ de las funciones aproximantes es el espacio de polinomios de grado como máximo n, es fácil verificar si se cumple la condición (3.9). Se puede hacer uso del hecho de que una función en $\mathscr{P}_n$ tiene como máximo n cambios de signos. Consecuentemente, si la función de error $[f(x) - p^*(x)]$ cambia de signo más de n veces mientras x oscila a lo largo de $\mathscr{Z}_M$, entonces p^* es la mejor aproximación. Por el contrario, si el número de cambios de signo no excede n, entonces podremos elegir los ceros de un cierto polinomio en $\mathscr{P}_n$ para que se cumpla la condición (3.9). Este resultado es el llamado teorema de caracterización.

Hay ciertas propiedades de los polinomios que conviene recordar, puesto que van a ser útiles en la demostración del teorema, estas son:

1. Si un elemento de $\mathscr{P}_n$ tiene más de n ceros, entonces es idénticamente cero.

2. Sea $\{\zeta_j : \ j = 1, 2, \ldots, k\}$ un conjunto cualquiera de puntos en el intervalo abierto (a, b), donde $k \leq n$. Entonces, existe un elemento de $\mathscr{P}_n$ que cambia de signo en estos puntos y que no tiene otros ceros. Además, existe una función en $\mathscr{P}_n$ que no tiene ceros en $[a, b]$.

3. Si una función definida en $\mathscr{P}_n$ que no es idénticamente nula tiene j ceros, y si k de esos ceros son puntos interiores de $[a, b]$ en los que la función no cambia de signo, entonces el número $(j + k)$ no es mayor que n.

4. Sea $\{\xi_j : \ j = 0, 1, \ldots, n\}$ un conjunto cualquiera de puntos en el intervalo $[a, b]$ y sea $\{\phi_j : \ j = 0, 1, \ldots, n\}$ una base cualquiera de $\mathscr{P}_n$. Entonces, la matriz $(n + 1) \times (n + 1)$ cuyos elementos tiene los valores $\{\phi_i(\xi_j) : \ i = 0, 1, \ldots, n; \ j = 0, 1, \ldots, n\}$ no es singular.

Definición 3.3.1. Un subespacio vectorial $\mathscr{A}$ de dimensión $n + 1$ de $\mathscr{C}[a, b]$ se dice que cumple la condición de Haar si las cuatro propiedades anteriores son ciertas, cuando $\mathscr{P}_n$ es reemplazado por el conjunto $\mathscr{A}$. Equivalentemente, cualquier base de $\mathscr{A}$ es llamada conjunto de Chebyshev.

Como las propiedades 1, 3 y 4 son equivalentes e implican la propiedad 2, es usual definir la condición de Haar, únicamente mediante la primera. Es decir, $\mathscr{A}$ satisface la condición de Haar si y solo si, para cada p en $\mathscr{A}$ no nulo, el número de raíces de la ecuación $\{p(x) = 0 : \ \ a \leq x \leq b\}$ es menor que la dimensión de $\mathscr{A}$.

Teorema 3.3.2 (Teorema de la Vallée Poussin)**.** *Sea $f \in \mathscr{C}[a,b]$ y $p^* \in \mathscr{P}_n$. Suponiendo que existen $n+2$ puntos $x_0 < \ldots < x_{n+1}$ en el intervalo $[a,b]$, tales que $f(x_i) - p^*(x_i)$ y $f(x_{i+1}) - p^*(x_{i+1})$ tiene signos contrarios para $i = 0, 1, \ldots, n$. Entonces,*

$$\min_{q \in \mathscr{P}_n} \|f - q\|_\infty \geq \min_{i=0,1,\ldots,n+1} |f(x_i) - p^*(x_i)|.$$

Demostración. La condición de los signos de $f(x_i) - p^*(x_i)$ se expresa normalmente diciendo que $f - p^*$ tiene los signos alternados en los puntos x_i, $i = 0, 1, \ldots, n+1$. Denotemos la expresión a la derecha de la desigualdad con μ. Claramente, $\mu \geq 0$: cuando $\mu = 0$, lo expuesto en el teorema es trivial, así que asumimos que $\mu > 0$. Por reducción al absurdo, supongamos que la desigualdad es falsa; entonces, para toda aproximación minimax $r_n \in \mathscr{P}_n$ a la función f, se obtiene lo siguiente:

$$\|f - r_n\|_\infty = \min_{q \in \mathscr{P}_n} \|f - q\|_\infty < \mu.$$

Por lo tanto,

$$|r_n(x_i) - f(x_i)| < |p^*(x_i) - f(x_i)|, \quad i = 0, 1, \ldots, n+1.$$

Ahora,

$$p^*(x_i) - r_n(x_i) = [p^*(x_i) - f(x_i)] - [r_n(x_i) - f(x_i)], \quad i = 0, 1, \ldots, n+1.$$

Ya que el primer término a la derecha siempre excede el segundo término en valor absoluto, se obtiene que $p^*(x_i) - r_n(x_i)$ y $p^*(x_i) - f(x_i)$ tiene el mismo signo para $i = 0, 1, \ldots, n+1$. Por tanto, $p^* - r_n$, es un polinomio de grado n y que cambia $n+1$ veces de signo, lo que es absurdo. Y la demostración queda acabada. □

Este último teorema nos da una pista para formular una caracterización constructiva del polinomio minimax; de hecho, debemos demostrar que, si las cantidades $|f(x_i) - p^*(x_i)|$, $i = 0, 1, \ldots, n+1$ del teorema anterior son iguales a $\|f - p^*\|_\infty$, entonces $p^* \in \mathscr{P}_n$ es un polinomio minimax de grado n para la función f en el intervalo $[a,b]$. Este resultado se conoce formalmente como el teorema de caracterización de la aproximación minimax, como formularemos a continuación.

De facto, si la función error $[f(x) - p^*(x)]$ cambia de signo $n+1$ o más veces cuando x recorre $\mathscr{Z}_M$, entonces p^* es una aproximación minimax de f. Recíprocamente, si el número de cambios de signo es menor o igual que n, entonces podemos elegir las abscisas donde se producen estos cambios de

signo como los ceros de un polinomio $p(x)$ de grado menor o igual que n que satisface la condición (3.9). La misma prueba vale para un subespacio $\mathcal{A}$ de dimensión $n+1$ que satisfaga la condición de Haar. En el argumento anterior es esencial la propiedad 2 enumerada en esta misma sección. La prueba que aportaremos, sin embargo, parte de primeros principios utilizando propiedades que son bien conocidas de los polinomios.

Teorema 3.3.3 (Teorema de caracterización)**.** *Sea f una función en $\mathscr{C}[a,b]$. Entonces, $p^* \in \mathcal{P}_n$ es la aproximación minimax a f en $\mathcal{P}_n$, si y solo si existen $n+2$ puntos $\{\xi_i^* : i = 0,1,2,\ldots,n+1\}$ que cumplan las siguientes condiciones:*

1. $a \leq \xi_0^* < \xi_1^* < \ldots < \xi_{n+1}^* \leq b$
2. $|f(\xi_i^*) - p^*(\xi_i^*)| = \|f - p^*\|_\infty, \quad i = 0,1,\ldots,n+1$
3. $f(\xi_{i+1}^*) - p^*(\xi_{i+1}^*) = -\left[f(\xi_i^*) - p^*(\xi_i^*)\right], \quad i = 0,1,\ldots,n.$

Demostración. Si $f \in \mathscr{P}_n$, entonces el resultado es cierto, con $p^* = f$ y cualquier sucesión de $n+2$ puntos distintos x_i, $i = 0,1,\ldots,n+1$ contenidos en $[a,b]$.

Nos falta probar, entonces, el caso en que $f \notin \mathscr{P}_n$, esto significa que no existe ningún polinomio $p \in \mathscr{P}_n$ que restringido en $[a,b]$ sea idénticamente igual a f. Para ver la suficiencia de la condición, suponemos que existe la secuencia de puntos $x_i, i = 0,1,\ldots,n+1$, con las propiedades dadas. Definimos,

$$L = \|f - p^*\|_\infty \;\; y \;\; E_n(f) = \min_{q \in \mathscr{P}_n} \|f - q\|_\infty.$$

Por el teorema de La Vallée - Poussin, se tiene que $E_n(f) \geq L$. Por definición de $E_n(f)$, obtenemos que $E_n(f) \leq \|f - p^*\|_\infty = L$. Con lo que llegamos a $E_n(f) = L$ y el polinomio dado p^* es un polinomio minimax.

Para la condición necesaria, suponemos que el polinomio dado $p^* \in \mathscr{P}_n$ es un polinomio minimax para f en $[a,b]$. Como $x \to |f(x) - p^*(x)|$ es una función continua en un intervalo cerrado y acotado $[a,b]$, existe un punto en $[a,b]$ en el que $|f(x) - p^*(x)|$ alcanza su máximo valor, $L > 0$; sea

$$x_0 = \text{mín}\left\{x \in [a,b] : |f(x) - p^*(x)| = L\right\}.$$

Ahora, $x_0 = b$ implicaría que $|f(x) - p^*(x)| = L$ para todo $x \in [a,b]$. Como f es continua en $[a,b]$, podremos decir que f es de las dos maneras siguientes: o bien $f(x) = p^*(x) + L$ para todo $x \in [a,b]$ o bien $f(x) = p^*(x) - L$ para todo $x \in [a,b]$. En cualesquiera de los casos encontraríamos que $f \in \mathscr{P}_n$, pero esto no puede ser, puesto que habíamos elegido $f \notin \mathscr{P}_n$. Por lo tanto, si $x_0 \in [a,b)$, y podemos asumir sin pérdida de generalidad que $f(x_0) - p^*(x_0) = L > 0$.

Ahora, debemos probar la existencia del siguiente punto crítico, $x_1 \in (x_0, b]$, tal que $f(x_1) - r(x_1) = -L$. Suponemos de otro modo, por contradicción que $-L < f(x) - p^*(x) \geq L$ para todo $x \in [a, b]$. Entonces, por la continuidad de f, existe un $\delta \in (0, L)$ tal que $-L + \delta \leq f(x) - r(x) \leq L$ para todo $x \in [a, b]$. Definimos entonces $r^* \in \mathscr{P}_n$ mediante

$$r^*(x)p^*(x) + \varepsilon,$$

donde $0 < \varepsilon < \min\{\delta, L\} = \delta$. Entonces, para todo $x \in [a, b]$,

$$f(x) - r^*(x) = f(x) - p^*(x) - \varepsilon \geq -L + \delta - \varepsilon > -L$$

y

$$f(x) - r^*(x) = f(x) - p^*(x) - \varepsilon \leq L - \varepsilon < L.$$

Las desigualdades anteriores significan que

$$\|f - r^*\|_\infty < K = \|f - r\|_\infty.$$

Por lo tanto, $r^* \in \mathscr{P}_n$ es una aproximación a f en $[a, b]$ mejor que $p^* \in \mathscr{P}_n$. Esto, sin embargo, contradice nuestra hipótesis de que p^* es un polinomio de mejor aproximación de f en $[a, b]$, lo que implica la existencia de

$$x_1 = \text{ínf}\{z \in (x_0, b] : f(x) - p^*(x) = -L\}.$$

En consecuencia, $f(x_1) - p^*(x_1) = -L$ y $x_1 \in (x_0, b]$, como se estaba buscando. Acabamos de probar para $n = 0$. Ahora, haciendo uso del método de inducción, suponemos que $n \geq 1$ y sucesivamente definiremos los puntos críticos como:

$$x_i = \text{ínf}\left\{x \in (x_{i-1}, b] : f(x) - p^*(x) = (-1)^i L\right\}, \quad i = 1, \ldots, m.$$

Esto continuará hasta que $x_m = b$ o hasta que encontremos un $x_m < b$, tal que $|f(x) - p^*(x)| < L$ para todo $x \in (x_m, b]$. Ahora, o bien $m \geq n + 1$ y, por tanto, acabaríamos con la demostración ya que habríamos encontrado $n + 2$ puntos críticos, o bien $1 \leq m \leq n$.

Para acabar la demostración del teorema, tendremos que probar que la segunda opción, es decir, $1 \leq m \leq n$ lleva a contradicción y, por tanto, no es posible. Supongamos que $1 \leq m \leq n$ y sea $\eta_0 = a$. Debido a la definición de los puntos x_i, $i = 0, 1, \ldots, m$,

$$\exists \eta_i \in (x_{i-1}, x_i)\ \forall x \in [\eta_i, x_i)\ |f(x) - p^*(x)| < L,\ i = 1, \ldots, m$$

y definimos $\eta_{m+1} = b$.

Se sigue de la elección de η_i, $i = 0, 1, \ldots, m + 1$, que se mantienen las siguientes propiedades:

1. $|f(x) - p^*(x)| \leq L$ para todo $x \in [\eta_i, \eta_{i+1}]$ y para todo $i = 0, 1, \ldots, m$.
2. para cada $i = 0, 1, \ldots, m$ existe $x_i \in [\eta_i, \eta_{i+1}]$ tal que $f(x_i) - p^*(x_i) = (-1)^i L$.
3. no existe un índice $i = 0, 1, \ldots, m$ y $x \in [\eta_i, \eta_{i+1}]$ tal que f$(x) - p^*(x) = (-1)^{i+1} L$.
4. $|f(\eta_i) - p^*(\eta_i) < L$ para todo $i = 1, \ldots, m$

Ahora, sea

$$v(x) = \prod_{i=1}^{m} (\eta_i - x)$$

y definimos

$$r^*(x) = r(x) + \varepsilon v(x)$$

donde $\varepsilon > 0$ es un número fijo real. Por la hipótesis $1 \leq m \leq n$, se sigue que $r^* \in \mathscr{P}_n$. Vamos a considerar el comportamiento de la diferencia:

$$f(x) - r^*(x) = f(x) - p^*(x) - \varepsilon v(x)$$

en cada uno de los intervalos $[\eta_i, \eta_{i+1}],\ i = 0, 1, \ldots, m$, cuya unión es $[a, b]$. Debemos probar que para $\varepsilon > 0$ lo suficientemente pequeño,

$$|f(X) - r^*(x)| < L = \|f - p^*\|_\infty$$

para todo x en $[\eta_i, \eta_{i+1}]$ con $i = 0, 1, \ldots, m$, esto es, $\|f - r^*\|_\infty < \|f - p^*\|_\infty$, lo que contradice el hecho de que $p^* \in \mathscr{P}_n$ es un polinomio minimax para f en $[a, b]$, lo que refuta la hipótesis de que $1 \leq m \leq n$.

Tomamos por ejemplo el intervalo $[\eta_0, \eta_1]$. Para cada x en $[\eta_0, \eta_1)$ se tiene que $v(x) > 0$ y, en consecuencia, por la definición de $r^*(x)$ y la propiedad (1) de arriba, se tiene:

$$f(x) - r^*(x) \leq L - \varepsilon v(x) < L, \quad x \in [\eta_0, \eta_1).$$

Además, como $v(\eta_1) = 0$, se sigue de la propiedad (4) que

$$f(\eta_1) - r^*(\eta_1) = f(\eta_1) - r(\eta_1) < L.$$

Por tanto, $f(x) - r^*(x) < L$ para cada x en $[\eta_0, \eta_1]$ y existe $\delta_1 \in (0, L)$ tal que $f(x) - p^*(x) \geq -L + \delta_1$ para todo x en $[\eta_0, \eta_1]$. Por lo que para $0 < \varepsilon < \text{mín}\,\{L, \delta_1, \varepsilon_1\}$, donde

$$\varepsilon_1 = \frac{\delta_1}{\text{máx}_{x \in [\eta_0, \eta_1]} |v(x)|}$$

tenemos que

$$f(x) - r^*(x) \geq -L + \delta_1 - \varepsilon|v(x)| > -L$$

Además, por la propiedad (4) anterior,

$$f(\eta_1) - r^*(\eta_1) = f(\eta_1) - p^*(\eta_1) > -L.$$

Por ello, $f(x) - r^*(x) > -L$ para todo $x \in [\eta_0, \eta_1]$, con $0 < \varepsilon < \text{mín}\{L, \delta_1, \varepsilon_1\}$. Combinando los límites superior e inferior en $f(x) - r^*(x)$, deducimos que

$$|f(x) - r^*(x)| < L = \|f - p^*\|_\infty, \quad x \in [\eta_0, \eta_1].$$

Argumentando de la misma manera en cada uno de los otros intervalos $[\eta_i, \eta_{i+1}]$, $i = 1, \ldots, m$ con $0 < \varepsilon < \text{mín}\{L, \delta_{i+1}, \varepsilon_{i+1}\}$, $i = 1, \ldots, m$, y δ_{i+1} y ε_{i+1} definidos análogamente a δ_1 y ϵ_1, concluimos que

$$|f(x) - r^*(x)| < L = \|f - p^*\|_\infty, \quad x \in [\eta_i, \eta_{i+1}], \;\; i = 0, 1, \ldots, m$$

y, por ello, para $0 < \varepsilon < \text{mín}\{L, \delta_1, \varepsilon_1, \ldots, , \delta_{m+1}, \varepsilon_{m+1}\}$,

$$\|f - r^*\|_\infty < L = \|f - r\|_\infty.$$

Como r^* está en $\mathscr{P}_n$, la última desigualdad contradice nuestra suposición de que p^* es un polinomio de la mejor aproximación a f en $[a, b]$ desde $\mathscr{P}_n$. La contradicción descarta la posibilidad de que $1 \leq m \leq n$. Ya que $m \geq 1$, se sigue que $m \geq n + 1$ y la demostración se acaba. □

En este teorema se ha supuesto, sin pérdida de generalidad, que $f(x_0) - p^*(x_0) = L > 0$, donde $L = \|f - p^*\|_\infty$. Cuando $f(x_0) - p^*(x_0) = -L < 0$ la prueba es análoga, excepto cuando definimos $r^* = p^* - \varepsilon$ para probar la existencia del punto crítico $x_1 \in (x_0, b]$ y en la discusión del caso $1 \leq m \leq n$, permitimos

$$r^*(x) = p^*(x) - \varepsilon v(x)$$

con $v(x)$ y $\varepsilon > 0$ definido como antes.

Una aplicación importante de este problema es para los polinomios de Chebyshev. El polinomio de Chebyshev, que denotaremos T_n, es un polinomio de grado n definido en el intervalo $[-1, 1]$ por la ecuación:

$$T_n(x) = \cos(n\theta), \quad x = \cos\theta, \quad 0 \leq \theta \leq \pi.$$

Pese a su forma, T_n es un polinomio de grado exactamente n. Se tiene que $T_0(x) = 1$, $T_1(x) = x$ para todo $x \in [-1, 1]$. Recordando la identidad trigonométrica

$$\cos(n+1)\theta + \cos(n-1)\theta = 2\cos\theta\cos n\theta$$

y fijando $\theta = \cos^{-1} x$, $\cos x \in [-1, 1]$ obtenemos la relación de recurrencia:

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x), \quad n = 1, 2, 3, \ldots, \quad x \in [-1, 1].$$

Se deduce que T_n es un polinomio de grado n en $[-1, 1]$.

Teorema 3.3.4. *Sea x definido en el intervalo $[-1, 1]$, y sea n cualquier entero positivo. El polinomio $\left(\frac{1}{2}\right)^{n-1} T_n$ pertenece a $\mathscr{P}_n$ y su norma infinito es la menor, bajo la condición de que el coeficiente de x^n sea igual a uno, es decir, que el polinomio de Chebyshev sea mónico.*

Demostración. Una manera de identificar el polinomio buscado es encontrando los valores de los coeficientes $\{c_i : \ i = 0, 1, \ldots, n-1\}$ que minimizan la siguiente expresión:

$$\max_{-1 \le x \le 1} \left| x^n + \sum_{i=0}^{n-1} c_i x^i \right|$$

Esto es equivalente a buscar la mejor aproximación de $\mathscr{P}_{n-1}$ a la función $\{x^n : \ -1 \le x \le 1\}$. Por el teorema de caracterización previo, se obtiene que $\left(\frac{1}{2}\right)^{n-1} T_n$ es el polinomio buscado, si el coeficiente de x^n es uno y si existen puntos $\{\xi_i : \ i = 0, 1, \ldots, n\}$ en $[-1, 1]$, ordenados de manera ascendente, de tal modo que se cumpla la siguiente ecuación:

$$T_n(\xi_i) = (-1)^{n-i} \|T_n\|_\infty, \quad i = 0, 1, \ldots, n. \tag{3.10}$$

La relación de recurrencia de los polinomios de Chebyshev:

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$$

implica que el coeficiente de x^n es el buscado, 1. Además por la definición $T_n(x) = \cos(n\theta)$ muestra que la ecuación (3.10) se satisface si permitimos a cada ξ_i tener el valor $\cos\left(\frac{(n-i)\pi}{n}\right)$. □

La principal razón por la que se permite a $\mathscr{X}$ un subconjunto cerrado de $\mathscr{C}[a, b]$ es por la necesidad en el algoritmo de intercambio de que $\mathscr{X}$ contenga únicamente $n + 2$ puntos. En la descripción del algoritmo llamaremos a estos puntos una referencia. Se permite que $\{\xi_i : \ i = 0, 1, \ldots, n+1\}$ sean los puntos de la referencia. Asumiremos siempre que estos puntos cumplen:

$$a \le \xi_0 < \xi_1 < \ldots < \xi_{n+1} \le b.$$

Teorema 3.3.5. *Sea $\mathscr{A}$ un subespacio vectorial de $\mathscr{C}[a,b]$ de dimensión $n+1$ que satisface la condición de Haar, sea $\{\xi_i : i = 0, 1, \ldots, n+1\}$ una referencia, y sea f una función cualquiera en $\mathscr{C}[a,b]$. Entonces, p^* es la función en $\mathscr{A}$ que minimiza la expresión*

$$\max_{i=0,1,\ldots,n+1} |f(\xi_i) - p^*(\xi_i)|, \quad p \in \mathscr{A} \tag{3.11}$$

si y solo si se cumple la siguiente ecuación

$$f(\xi_{i+1}) - p^*(\xi_{i+1}) = -[f(\xi_i) - p^*(\xi_i)], \quad i = 0, 1, \ldots, n. \tag{3.12}$$

Demostración. Este teorema es una aplicación directa del teorema de caracterización. Exceptuando el hecho de que permitimos a $\mathscr{Z}$ ser el conjunto de puntos $\{\xi_i : i = 0, 1, \ldots, n+1\}$, en vez del intervalo $[a,b]$. □

La función p^* que minimiza la primera expresión del teorema, es decir, $\max_{i=0,1,\ldots,n+1} |f(\xi_i) - p(\xi_i)|$ puede ser calculada gracias a las ecuaciones $f(\xi_{i+1}) - p^*(\xi_{i+1}) = -[f(\xi_i) - p^*(\xi_i)]$. Normalmente se denomina h al valor $[f(\xi_0) - p^*(\xi_0)]$, y una vez elegida una base de $\mathscr{A}$, $\{\phi_j : \ j = 0, 1, \ldots, n\}$. Los coeficientes de la función

$$p^*(x) = \sum_{j=0}^{n} \lambda_j \phi_j(x), \quad a \leq x \leq b$$

satisfacen las ecuaciones:

$$f(\xi_i) - \sum_{j=0}^{n} \lambda_j \phi_j(\xi_i) = (-1)^i h, \quad i = 0, 1, \ldots, n+1.$$

Estas son un sistema lineal con las incógnitas $\{\lambda_j : \ j = 0, 1, \ldots, n\}$ y h. Por el teorema previo, obtenemos que estas ecuaciones tienen una solución para todas las funciones f en $\mathscr{C}[a,b]$, la matriz del sistema es no singular. Por lo tanto, únicamente existe un elemento de $\mathscr{A}$ que minimice la expresión $\max_{i=0,1,\ldots,n+1} |f(\xi_i) - p(\xi_i)|$. Un método más general para probar la unicidad se dará en la siguiente sección.

3.4. Unicidad y cotas del error minimax

Durante toda esta sección vamos a considerar que las condiciones del eorema de caracterización se cumplen; recordándolas son:

1. $a \leq \xi_0^* < \xi_1^* < \ldots < \xi_{n+1}^* \leq b$

2. $|f(\xi_i^*) - p^*(\xi_i^*)| = \|f - p^*\|_\infty, \quad i = 0, 1, \ldots, n+1$

3. $f(\xi_{i+1}^*) - p^*(\xi_{i+1}^*) = -\left[f(\xi_i^*) - p^*(\xi_i^*)\right], \quad i = 0, 1, \ldots, n$

Sean además p^* y q^* las mejores aproximaciones de $\mathscr{A}$ a f y tales que se cumplen las condiciones anteriores para ambas. Sea r^* la función $(q^* - p^*)$, consideramos:

$$r^*(\xi_i^*) = [f(\xi_i^*) - p^*(\xi_i^*)] - [f(\xi_i^*) - q^*(\xi_i^*)], \quad i = 0, 1, \ldots, n+1.$$

Como $\|f - q^*\|_\infty$ y $\|f - r^*\|_\infty$ son iguales, se obtiene por la condición (2) del teorema de caracterización que o bien $r^*(\xi_i^*)$ es cero, o bien su signo es el mismo que $[f(\xi_i^*) - p^*(\xi_i^*)]$. Gracias a la condición (3) del teorema de caracterización, se obtiene información sobre los signos de los términos de la secuencia $\{r^*(\xi_i^*) : \ i = 0, 1, \ldots, n+1\}$. De esto se deduce que r^* es idénticamente cero. Por lo tanto, la proximación de $\mathscr{A}$ a f es única.

Teorema 3.4.1. *Sea r una función en $\mathscr{C}[a, b]$, y sea $\{\xi_i : \ i = 0, 1, \ldots, n+1\}$ una referencia, tal que se satisfacen las condiciones*

$$(-1)^i r(\xi_i) \geq 0, \quad i = 0, 1, \ldots, n+1 \tag{3.13}$$

Entonces r tiene al menos $(n+1)$ ceros en $[a, b]$ contando cada cero doble dos veces, donde un cero doble es un cero que se encuentra estrictamente dentro de $[a, b]$ y donde r no cambia de signo.

Demostración. Sean $\mathcal{I}$ y $\mathcal{J}$ los conjuntos definidos como:

$$\mathcal{I} = \{i : \ r(\xi_i) \neq 0, \quad i = 0, 1, \ldots, n+1\}$$

$$\mathcal{J} = \{j : \ r(\xi_j) = 0, \quad j = 0, 1, \ldots, n+1\}$$

y sean $n(\mathcal{I})$ y $n(\mathcal{J})$ el número de elementos en cada conjunto.

El teorema es trivial si $n(\mathcal{I})$ es cero o uno.

En caso contrario, se considera el número de ceros en el intervalo $[\xi_k, \xi_l]$ donde k y l se encuentran ambos en $\mathcal{I}$, y ningún otro elementos de $\mathcal{I}$ está en $[k, l]$. La condición $(-1)^i r(\xi_i) \geq 0$ implica que los números $r(\xi_k)$ y $r(\xi_l)$ tienen el mismo signo si $(l - k)$ es par, y tienen el signo contrario si $(l - k)$ es impar. Por lo tanto, el número de ceros de r en el intervalo $[\xi_k, \xi_l]$ es, al menos, uno más que el número de puntos del conjunto $\{\xi_j : j \in \mathcal{J}\}$ que se encuentra en este intervalo, lo que prueba que cualquier doble cero se cuenta dos veces, ya que el número de parejas $[\xi_k, \xi_l]$ que tienen esta propiedad es $n(\mathcal{I}) - 1$. Se deduce que el número total de ceros de r en $[a, b]$ es al menos $n(\mathcal{I}) + n(\mathcal{J}) - 1$. □

Teorema 3.4.2. *Sea $\mathscr{A}$ un subespacio vectorial en $\mathscr{C}[a,b]$ que satisface la condición de Haar. Entonces, para cualquier f en $\mathscr{C}[a,b]$, la mejor aproximación de $\mathscr{A}$ a f existe y es única.*

Demostración. Supongamos que existen dos mejores aproximaciones que denotaremos como p^* y q^*, entonces la función $(p^* - q^*)$ tendrá al menos $n+1$ ceros en $[a,b]$, siempre contando los ceros dobles dos veces. Recordando la propiedad de los polinomios, que se cumple siempre y cuando se tenga la condición de Haar en que, si una función de $\mathscr{P}_n$ que no es idénticamente cero tiene j ceros y k de los cuales son puntos interiores de $[a,b]$ en los que la función no cambia de signo, entonces el número $(j+k)$ no puede ser mayor que n. Debido a ella, las funciones p^* y q^* son las mismas. □

Teorema 3.4.3. *Si se mantienen las condiciones del teorema de caracterización, sea p^* cualquier elemento de $\mathscr{A}$ y sea $\{\xi_i : \ i = 0, 1, \ldots, n+1\}$ una referencia, que satisface la siguiente condición:*

$$sign[f(\xi_{i+1}) - p^*(\xi_{i+1})] = -sign[f(\xi_i) - p^*(\xi_i)], \quad i = 0, 1, \ldots, n.$$

Entonces se cumplen las siguientes desigualdades:

$$\begin{aligned} \min_{i=0,1,\ldots,n+1} |f(\xi_i) - p^*(\xi_i)| &\leq \min_{p\in\mathscr{A}} \max_{i=0,1,\ldots,n+1} |f(\xi_i) - p(\xi_i)| \qquad (3.14)\\ &\leq \min_{p\in\mathscr{A}} \|f - p\|_\infty \qquad (3.15)\\ &\leq \|f - p\|_\infty \qquad (3.16) \end{aligned}$$

Además, la primera desigualdad es estricta a no ser que todos los números $\{|f(\xi_i) - p^*(\xi_i)| : \ i = 0, 1, \ldots, n+1\}$ sean iguales.

Demostración. Para ver la primera desigualdad, suponemos que existe una función q^* en $\mathscr{A}$ que satisface la condición:

$$\min_{i=0,1,\ldots,n+1} |f(\xi_i) - p^*(\xi_i)| \geq \max_{i=0,1,\ldots,n+1} |f(\xi_i) - q^*(\xi_i)|$$

Si q^* es igual a p^*, entonces la expresión muestra que los números

$$\{|f(\xi_i) - p^*(\xi_i)| : \ i = 0, 1, \ldots, n+1\}$$

son todos iguales. Y se seguiría cumpliendo la desigualdad. Ahora, supongamos que p^* no es igual a q^*, pero que se satisfaga la condición

$$\min_{i=0,1,\ldots,n+1} |f(\xi_i) - p^*(\xi_i)| \geq \max_{i=0,1,\ldots,n+1} |f(\xi_i) - q^*(\xi_i)|$$

Denotemos a r^* como la función $(q^* - p^*)$. Por la condición anterior, se tiene que los números $r^*(\xi_i^*) = [f(\xi_i^*) - p^*(\xi_i^*)] - [f(\xi_i^*) - q^*(\xi_i^*)]$ tienen las mismas propiedades del signo que antes. Por el teorema 3.4.1 y por la condición de Haar, se obtiene que las funciones p^* y q^* deben ser las mismas, con lo que obtenemos una contradicción y la primera desigualdad queda probada. La segunda desigualdad se cumple porque la referencia es un subconjunto de $[a, b]$. La tercera desigualdad de la expresión se mantiene debido a que p^* pertenece a $\mathscr{A}$. □

Se debe notar, que si p^* es la mejor aproximación minimax de $\mathscr{A}$ a f y si la referencia en el último teorema son los puntos $\{\xi_i^* : i = 0, 1, \ldots, n+1\}$ que se dan con las condiciones del teorema de caracterización, entonces, todas las desigualdades del teorema anterior se satisfacen como ecuaciones.

3.5. Algoritmos

La computación de una mejor aproximación en un intervalo (o en cualquier otro espacio métrico compacto) puede ser llevada a cabo mediante sucesiones de las mejores aproximaciones en conjuntos cada vez más finos. Para estos problemas discretos son útiles los algoritmos explicados en el capítulo anterior. Para buscar una aproximación $\sum_{j=1}^n c_j g_j$ de una función f en un conjunto X, se debe seleccionar un conjunto finito $Y = \{x_1, \ldots, x_n\}$ y minimizar la función

$$\Delta(c_1, \ldots, c_n) = \max_{1 \le i \le m} |\sum_{j=1}^{n} c_j g_j(x_i) - f(x_i)|$$

Esto es lo mismo que resolver en sentido minimax los siguientes sistemas de ecuaciones lineales:

$$\sum_{j=1}^{n} c_j g_j(x_i) = f(x_i) \ \ (i = 1, \ldots, m).$$

Para el algoritmo se requiere únicamente que las funciones $\{f, g_1, \ldots, g_n\}$ sean continuas en el espacio métrico compacto X. El problema es determinar un vector de coeficientes $\mathbf{x} = (c_1, \ldots, c_n)$ para el que

$$\Delta(c) = \| \sum_{i=1}^{n} c_i g_i - f \| = \max_{x \in X} | \sum_{i=1}^{n} c_i g_i - f |$$

sea mínimo. Si el conjunto $\{g_1, \ldots, g_n\}$ no es independiente, se podrá reemplazar por un conjunto más pequeño que sí lo sea, sin incrementar $p = \text{ínf } \Delta(c)$. Se definen las funciones de residuo como $r(c, \mathbf{x}) = \sum c_i g_i(\mathbf{x}) - f(\mathbf{x})$.

3.5.1. El algoritmo de intercambio

3.5.1.1. Resumen del algoritmo de intercambio

Sea f una función definida en $\mathscr{C}[a,b]$ y sea un subespacio vectorial de dimensión $n+1$ $\mathscr{A}$ de $\mathscr{C}[a,b]$ que satisface la condición de Haar. El algoritmo de intercambio calcula el elemento de $\mathscr{A}$ que minimiza el máximo error, es decir,

$$\|f-p\|_\infty = \max_{a\le x\le b} |f(x)-p(x)|, \quad p\in\mathscr{A}. \tag{3.17}$$

En vez de intentar reducir el error en cada intento de aproximación, el algoritmo configura una referencia $\{\xi_i \ : i=0,1,\ldots,n+1\}$, con la que se converge em un conjunto de puntos $\{\xi_i^* \ : i=0,1,\ldots,n+1\}$ que satisface las siguientes condiciones del teorema de caracterización, que recordemos son:

- $a\le \xi_0^* < \xi_1^* < \ldots < \xi_{n+1}^* \le b$
- $|f(\xi_i^*)-p^*(\xi_i^*)| = \|f-p^*\|_\infty, \quad i=0,1,\ldots,n+1$
- $f(\xi_{i+1}^*)-p^*(\xi_{i+1}^*) = -\left[f(\xi_i^*)-p^*(\xi_i^*)\right], \quad i=0,1,\ldots,n.$

Las distintas modificaciones de la referencia se llevarán a cabo mediante un proceso iterativo.

Para empezar los cálculos, primero se debe elegir una referencia inicial. Puede ser cualquier conjunto de puntos que garanticen la condición

$$a\le \xi_0 < \xi_1 < \ldots < \xi_{n+1} \le b,$$

una elección particular que es adecuada cuando $\mathscr{A}$ es un espacio $\mathscr{P}^n$ se dará más adelante. Al principio de cada iteración se tendrá una referencia distinta a todas las referencias de las iteraciones anteriores. Los cálculos serán los siguientes.

Consideramos $\{\xi_i : \ i=1,\ldots,n+1\}$ la referencia en cada empiece de iteración. Primero, se debe calcular la función p en $\mathscr{A}$ que minimice la expresión

$$\max_{i=0,1,\ldots,n+1} |f(\xi_i)-p(\xi_i)|, \quad p\in\mathscr{A}$$

Hemos visto, en la sección anterior que los coeficientes de p pueden ser calculados al resolver el siguiente sistema lineal de ecuaciones:

$$f(\xi_i)-p(\xi_i) = (-1)^i h, \quad i=0,1,\ldots,n+1$$

Se satisfacen los siguientes límites, donde p^* es la mejor aproximación de $\mathscr{A}$ a f

$$|h| \le \|f-p^*\|_\infty \le \|f-p\|_\infty$$

Usando estos límites y utilizando el límite de la derecha, para poder obtener un cambio adecuado para la referencia, se considera la siguiente función de error:

$$e(x) = f(x) - p(x), \quad a \leq x \leq b$$

Distintos métodos eficaces se basan en encontrar puntos que encajen con la función de error de manera cuadrática, pero se asume que las abscisas de los extremos se pueden encontrar de manera exacta. Sea η un punto que satisfaga la ecuación

$$|f(\eta) - p(\eta)| = \|f - p\|_\infty$$

Los cálculos acabarán si la diferencia

$$\delta = |f(\eta) - p(\eta)| - |h|$$

es suficientemente pequeña, ya que por una desigualdad vista anteriormente, se tiene

$$\|f - p\|_\infty \leq \|f - p^*\| + \delta.$$

Si no se cumpliera, la referencia se cambiaría para poder empezar otra iteración. La propiedad más importante del cambio de referencia es que la cantidad $|h|$, llamada la referencia de error nivelada, debe crecer monótonamente de una iteración a otra.

Al pensar en la referencia de error nivelada como una función de referencia, se usa la siguiente notación: $h(\xi_0, \xi_1, \ldots, \xi_{n+1}) = |h|$. Es útil saber que el propósito del cambio de referencia es aumentar el valor de $h(\xi_0, \xi_1, \ldots, \xi_{n+1})$. La expresión de δ se hace pequeña solo si la referencia de error nivelada es cercana a $\|f - p^*\|_\infty$, por lo que el algoritmo de intercambio será un método para resolver el problema de maximización, donde las variables son los puntos de la referencia.

3.5.1.2. Ajustes en la referencia

Vamos a considerar una iteración del algoritmo de intercambio que calcula una función p de $\mathscr{A}$, resolviendo las ecuaciones:

$$f(\xi_i) - p(\xi_i) = (-1)^i h, \quad i = 0, 1, \ldots, n+1.$$

Esto cambia la referencia $\{\xi_i \ : i = 1, \ldots, n+1\}$ por $\left\{\xi_i^+ \ : i = 1, \ldots, n+1\right\}$. El método de elegir una nueva referencia depende del teorema 3.4.3, ya que implica el crecimiento:

$$h(\xi_0^+, \xi_1^+, \ldots, \xi_{n+1}^+) > h(\xi_0, \xi_1, \ldots, \xi_{n+1}).$$

El teorema muestra que es suficiente si se satisfacen las condiciones:

$$sign[f(\xi_{i+1}^+) - p(\xi_{i+1}^+)] = -sign[f(\xi_i^+) - p(\xi_i^+)] \quad i = 0, 1, \ldots, n+1 \tag{3.18}$$

$$|f(\xi_i^+) - p(\xi_i^+)| \geq |h|, \quad i = 0, 1, \ldots, n+1. \tag{3.19}$$

Esto nos muestra que, al menos, uno de los números

$$\left\{|f(\xi_i^+) - p(\xi_i^+)| : \ i = 0, 1, \ldots, n+1\right\}$$

es mayor que $|h|$.

Un método para obtener crecimiento en la referencia de error nivelada es permitir que cada punto de la nueva referencia sea un extremo de la función de error. Los métodos que cambian todos los puntos de la referencia son normalmente más eficientes que aquellos en los que solo se cambia un punto, ya que se requieren menos iteraciones para llegar a la tolerancia buscada.

3.5.1.3. Aplicaciones de los polinomios de Chebyshev en aproximación minimax

Una propiedad del algoritmo de intercambio es que, si la condición de Haar se mantiene, la convergencia puede ser obtenida desde cualquier referencia inicial. Algunas referencias iniciales son mejores que otras, puesto que se puede evitar el cálculo de aproximaciones con errores mayores de lo necesario. Cuando $\mathscr{A}$ es el espacio $\mathscr{P}_n$ una referencia inicial efectiva, puede ser obtenida gracias a las propiedades de los polinomios de Chebyshev. Específicamente si x pertenece al intervalo $[-1, 1]$, los puntos de la referencia inicial se toman de la siguiente manera:

$$\xi_i = \cos\left(\frac{(n+1-i)\pi}{n+1}\right), \quad i = 0, 1, \ldots, n+1 \tag{3.20}$$

ya que esta elección nos asegura la siguiente propiedad.

Teorema 3.5.1. *Sean $f \in \mathscr{C}[-1, 1]$ y $p \in \mathscr{P}_n$ con p la aproximación de f calculada con el algoritmo de intercambio, y con la referencia conteniendo los puntos definidos como en (3.20). Si f es un polinomio de grado $n+1$, entonces p es la mejor aproximación minimax de $\mathscr{P}_n$ a f.*

Demostración. La manera en la que están definidos los puntos de la referencia y la definición del polinomio de Chebyshev T_{n+1} implican los valores:

$$T_{n+1}(\xi_i) = (-1)^{n+1-i}, \quad i = 0, 1, \ldots, n+1.$$

Como $(f - p)$ pertenece a $\mathscr{P}_{n+1}$, se sigue de la ecuación $f(\xi_i) - p(\xi_i) = (-1)^i h$ que el error de la función $(f - p)$ es un múltiplo de T_{n+1}, por lo tanto, por el teorema de caracterización, p es la mejor aproximación de $\mathscr{P}_n$ a f. □

Este teorema es útil, no solo para cuando f pertenece a $\mathscr{P}_{n+1}$, sino también cuando f es infinitamente diferenciable y su serie de Taylor:

$$f(x) = \sum_{j=0}^{\infty} \frac{x^j}{j!} f^{(j)}(0), \quad i = 0, 1, \ldots, n+1$$

converge rápidamente. En este caso, a menudo, ocurre que el error de la mejor aproximación de $\mathscr{P}_n$ a f está dominado por el error del término $\frac{x^{n+1} f^{(n+1)}(0)}{(n+1)!}$. El teorema previo muestra que la referencia (3.20) ayuda a que el error sea lo más pequeño posible.

La referencia de puntos (3.20) es apropiada únicamente para el intervalo $[-1, 1]$. Para el intervalo general $[a, b]$ se debe tomar la siguiente referencia:

$$\xi_i = \frac{1}{2}(a+b) + \frac{1}{2}(b-a)\cos\left(\frac{(n+1-i)\pi}{(n+1)}\right), \quad i = 0, 1, \ldots, n+1.$$

3.5.2. Primer algoritmo de Remez

El algoritmo de Remez fue publicado por E. Y. Remez en 1934. Se trata de un algoritmo iterativo usado para encontrar aproximaciones minimax. Para inicializar el algoritmo, se necesita un conjunto de $n+2$ puntos en el intervalo $[a, b]$. Se pueden elegir de distintas formas, pero la más común es los nodos de Chebyshev. Esto es debido a que estos nodos evitan el fenómeno de Runge y minimizan el error de interpolación. Por tanto, el polinomio que pasa a través de los nodos de Chebyshev es una buen punto de inicio para el polinomio minimax. Consideremos que el polinomio $P_n(x)$ que pasa por los nodos de Chebysev es:

$$P_n(x) = b_0 + b_1 x^1 + \ldots + b_n x^n$$

donde $b_0, b_1, \ldots, b_n$ son los coeficientes. Ahora, queremos forzar el criterio de oscilación, es decir, que el error entre el polinomio y la función f oscilen alternativamente en los nodos Chebyshev. Para ello, escribimos el siguiente sistema de ecuaciones:

$$b_0 + b_1 x_i^1 + \ldots + b_n x_i^n + (-1)^i E = f(x_i) \quad i = 0, 1, 2, \ldots, n+1.$$

Se puede escribir esto de forma matricial como:

$$\begin{pmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n & E \\ 1 & x_1 & x_1^2 & \cdots & x_1^n & -E \\ 1 & x_2 & x_2^2 & \cdots & x_2^n & E \\ \vdots & \vdots & \vdots & \ddots & \vdots & \\ 1 & x_{n+1} & x_{n+1}^2 & \cdots & x_{n+1}^n & (-1)^{n+1}E \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ b_{n+1} \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f(x_1) \\ f(x_2) \\ \vdots \\ f(x_{n+1}) \end{pmatrix} \quad (3.21)$$

Podemos resolver este sistema $(n+2)(n+2)$ para encontrar los coeficientes $b_0, b_1, \ldots, b_n, E$. Hemos impuesto el criterio de oscilación, pero se debe notar que el error E no es necesariamente el extremo de la función de error. Por esta razón, debemos encontrar un nuevo conjunto de puntos. Esto nos lleva al segundo paso del algoritmo, denominado el paso del intercambio.

Lo que tenemos hasta ahora es una función de error que alterna el signo en los $n+2$ puntos. Haciendo uso del teorema del valor intermedio, se obtiene que la función tiene $n+1$ raíces. Hallamos las raíces usando cualquier método numérico y considerando los $n+2$ intervalos siguientes:

$$[a, z_0], [z_0, z_1], [z_1, z_2], \ldots, [z_{n-1}, z_n], [z_n, b]$$

donde $z_0, z_1, \ldots, z_n$ son las $n+1$ raíces. Para cada intervalo de los definidos arriba, encontramos el punto en que la función de error alcance su valor máximo o mínimo. Podemos hallarlo derivando la función de error y localizando el mínimo o el máximo en cada intervalo. Si sucede que el mínimo o el máximo no existen, calculamos el valor del error en los dos extremos del intervalo y tomamos aquel que tenga el mayor valor absoluto. Esto nos aporta un nuevo conjunto de puntos

$$x_0^*, x_1^*, \ldots, x_{n+1}^*$$

Este nuevo conjunto de puntos será usado en la segunda etapa de la iteración. Seguimos con la iteración hasta que se alcance el criterio de parada.

Al final de cada iteración, obtenemos un nuevo conjunto de puntos en los que evaluamos el error. Sea $E_m = \text{mín}_i |E_i|$ y $E_M = \text{máx}_i |E_i|$. Con el paso de las iteraciones, el algoritmo converge y se acerca al polinomio minimax, así pues la diferencia entre la antigua y la nueva referencia de puntos se va minimizando. Un criterio de parada razonable sería para las iteraciones cuando $E_M = \alpha E_m$ donde α es alguna constante cercana a 1.

En resumen, los pasos más importantes serían:

1. Para una función dada $f(x)$ en un intervalo $[a, b]$, se especifica el grado del polinomio de interpolación.

2. Se calculan los $n+2$ nodos de Chebyshev.

3. Imponiendo el criterio de oscilación, se resuelve el sistema de ecuaciones $(n+2) \times (n+2)$ para obtener $b_0, b_1, \ldots, b_n, E$.

4. Formamos un nuevo polinomio P_n con los coeficientes hallados $P_n(x) = b_0 + b_1 x^1 + \ldots + b_n x^n$.

5. Calculamos los extremos de la función de error $P_n(x) - f(x)$ y esto nos dará una nueva refenrencia $x_0^*, x_1^*, \ldots, x_{n+1}^*$.

6. Si se alcanza el criterio de parada, se paran los cálculos, si no, se usa la nueva referencia y se repite a partir del paso (3).

3.6. Código en MATLAB

En la implementación del algoritmo utilizaremos la función Matlab `fzero` para determinar indistintamente los ceros de la función error

$$e(x) = f(x) - P_n(x)$$

en el intervalo $[a, b]$, y los ceros de la derivada $e'(x)$, que determinarán los extremos de la función, que constituirán la nueva referencia para iniciar una nueva iteración.

El programa Matlab que sigue está compuesto de una rutina principal

$$\texttt{function[iter, y, B]} = \texttt{remez(fun, fun\_der, a, b, orden)}$$

a la que se da como argumentos la función `fun` que se desea aproximar, su derivada `fun_der`, el intervalo $[a, b]$, y el grado del polinomio de aproximación. Esta rutina llama internamente a una función que sirve para evaluar indistintamente en un array `x` la función error $e(x) = f(x) - P_n(x)$, o su derivada $e'(x) = f'(x) - P'_n(x)$. Nótese que el polinomio $P_n(x)$ se expresa en la base formada por las sucesivas potencias $(x - a)^k, k = 0, \ldots, n$.

```
1   % Algoritmo de Remez
2
3   function [iter,y, B1]=remez(fun,fun_der,a,b,orden)
4
5   %fun: función de la que buscamos su aproximación
6   %fun_der: derivada de la función
7   %a: extremo inferior del intervalo
8   %b: extremo superior del intervalo
9   %orden: grado del polinomio
10
11  format long
12  options = optimset('TolX',1.0e-9,'TolFun',1.0e-9);
13  tol = 1.0e-9;
14  maxiter = 1000;
15
16  potencia = ones(orden+2,1)*([0:orden]); %se trata de
        una matriz de potencias repetidas en filas (order +2)
         veces
17  coeff_E = (-1).^[1:orden+2]';   %se colocan los
        coeficientes de E como un vector columna
```

```
18        t = (1:orden)'; % las potencias del polinomio colocadas
              en una columna
19
20        % llevamos a cabo la primera elección de las abscisas
              mediante chebyshev
21        % para que sea más eficiente
22        k = 1:orden+2;
23        y = (1/2)*(a+b)-(1/2)*(b - a)*cos((2*k-1)*pi/(2*(orden
              +2))); % elegidos en el intervalo
24
25        errormax = 10; % valor aleatorio para que entre en el
              bucle
26        iter = 0;
27        disp(y);
28
29        while (errormax > tol*max(abs(a),abs(b)) && (iter <
              maxiter))
30
31        y = y(:); % ponemos en una columna las abscisas
32        % creamos la matriz
33        h = (y - a)*ones(1, orden + 1); % repetimos los puntos
              menos el principio del intervalo
34        % (order +1) veces
35
36        coeff_h = h.^potencia;    % eleva la matriz h a la
              potencias
37        M = [coeff_h coeff_E];    % matriz del sistema lineal de
              ecuaciones
38
39        F = feval(fun,y); % vector de los f(x)
40
41        B = M\F;   % solución del sistema lineal de ecuaciones,
              primero hay (orden+1) elementos
42        % que son los coeficientes del polinomio. El último
              elemento es
43        % el valor del error en esos puntos
44
45        B1 = B(1:end-1); % Cogemos solo los coeficientes
46        B_der = B(2:end-1).*t;    % coeficientes de la derivada
47
48        % Vamos ahora a ir elegiendo los orden+2 intervalos
49
50        z(1) = a;   % z(1) es el punto de inicio del intervalo
51        z(orden+3) = b;    % z(orden+3) es el punto final del
              intervalo
52        % entre medio se rellenan con las raíces de la función
              error.
53        for k = 1: orden+1
54        z(k+1) = fzero(@(x) err(x,fun,B1,a), [y(k), y(k+1)],
```

```
        options);
55      end
56
57      %entre cada dos puntos del array z, se debe buscar el
            punto que
58      %maximice la magnitud de la función de error.
59      %Si hay un extremo (maximo local o mínimo local) entre
            los dos puntos
60      %del array z, entonces la derivada de la función será
            cero en este
61      %extremo. Se debe buscar los puntos extremos mirando
            las raíces de la
62      %derivada de la función de error entre los dos puntos.
63      %Si el extremo no existe, entonces comprobaremos el
            valor de la
64      %función de error en los dos puntos acuales de z y
            eligiremos aquel
65      %con mayor magnitud.
66
67      for k = 1:orden+2
68      %si cambia de signo, buscamos el extremo y el valor del
             la función
69      %error en ese punto
70      if sign(err(z(k),fun_der,B_der,a))~=sign(err(z(k+1),
            fun_der,B_der,a))
71      y1(k) = fzero(@(x) err(x, fun_der, B_der, a),[z(k),z(k
            +1)],options);
72      v(k) = abs(err(y1(k),fun,B1,a));
73      else
74      %si no hay cambio de signo, quiere decir que no hay
            extremo y
75      %comparamos los puntos del intervalo
76      v1 = abs(err(z(k),fun,B1,a)); %magnitud de la función
            de error al principio del subintervalo
77      v2 = abs(err(z(k+1),fun,B1,a)); %magnitud de la función
             de error al final del subintervalo
78      %elegimos el que sea mayor
79      if v1 > v2
80      y1(k) = z(k);
81      v(k) = v1;
82      else
83      y1(k) = z(k+1);
84      v(k) = v2;
85      end
86      end
87      end
88
89      %buscamos el punto en el array de extremos que tenga la
             mayor magnitud
```

```
90        % de la función de error. Si la diferencia entre este
              punto y el
91        % correspondiente en el antiguo array es menor que una
              cierta
92        % tolerancia, salimos del bucle
93
94        errormax = max(abs(y-y1'));
95
96        % reemplazamos la referencia
97        y = y1;
98        disp(y);
99        iter = iter+1;
100       end
101       end
```

```
1         function e = err(x,fun,A,a)
2         % función del error
3         % x: punto donde vamos a calcular el error
4         % fun : la función de la que queremos sacar el error
5         % A: los coeficientes del polinomio de aproximación
6         % a: primer elemento de los nodos
7
8         % los coeficientes del polinomio les colocamos en una
              columna
9         A = A(:);
10        x = x(:);
11
12        % el orden del polinomio es igual al numero de
              coeficientes menos uno
13        order = length(A)-1;
14
15
16        % las potencias colocadas en una fila y repetidas para
              cada argumento para
17        % formar una matriz, por ejemplo si tenemos 3 elementos
              en x, entonces
18        %              [0 1 2]
19        % potencias = [0 1 2]
20        %              [0 1 2]
21
22        potencias = ones(length(x),1)*[0:order];
23
24        % cada argumento se repite un número de veces igual al n
              úmero de
25        % coeficientes que forman una fila entonces cada
              elemento de la fila
26        % resultante se alcanza con la correspondiente potencia
              en la matriz de
27        % potencias.
```

```
28
29        temp =((x-a)*ones(1,order+1)).^potencias;
30
31        %multiplicando la matriz resultante con la tabla de
              coeficientes para
32        %obtener un vector columna. Cada elemento del vector
              resultante es igual al
33        %polinomio evaluado en la distancia entre los
              coerrespondientes argumentos
34        %y el principio del intervalo
35
36        temp = temp*A;
37
38        %el vector error esta entonces dado como la diferencia
              entre la función
39        %evaluada en el punto dadp y el polinomio evaluado en
              ese mismo punto
40
41        e = feval(fun,x)-temp;
```

3.6.1. Problemas test

Ilustramos la efectividad del algoritmo de Remez con varios ejemplos, en los que el criterio de parada impone que el máximo de la diferencia entre dos referencias consecutivas debe ser mayor que una cierta tolerancia.

3.6.1.1. Ejemplo 1

Consideramos la aproximación minimax por polinomios cuadráticos de la función $f(x) = sin(x)$, en el intervalo $[0, \pi/2]$. La convergencia se alcanza después de cinco iteraciones y se ilustra en la tabla siguiente, que muestra las sucesivas referencias computadas por el algoritmo.

y_1	y_2	y_3	y_4
0.059784875362591	0.484839298455275	1.085957028339621	1.511011451432306
0	0.396351057488408	1.115913622125215	1.570796326794897
0	0.361226116017639	1.134168420088915	1.570796326794897
0	0.361145036766295	1.133338703212475	1.570796326794897
0	0.361145396685376	1.133338825666037	1.570796326794897
0	0.361145396685357	1.133338825665943	1.570796326794897

Cuadro 3.1: tabla de referencias

La convergencia cuadrática se manifiesta en la aparente duplicación del número de cifras coincidentes entre iterantes sucesivos.

El polinomio de aproximación será:

$$\begin{aligned} B_0 + x(B_1 + xB_2) &= -0{,}013864950803157 \\ &\quad +x(1{,}174881001423768 - x \cdot 0{,}331429235303895) \end{aligned}$$

La figura 3.1. muestra el aproximante minimax.

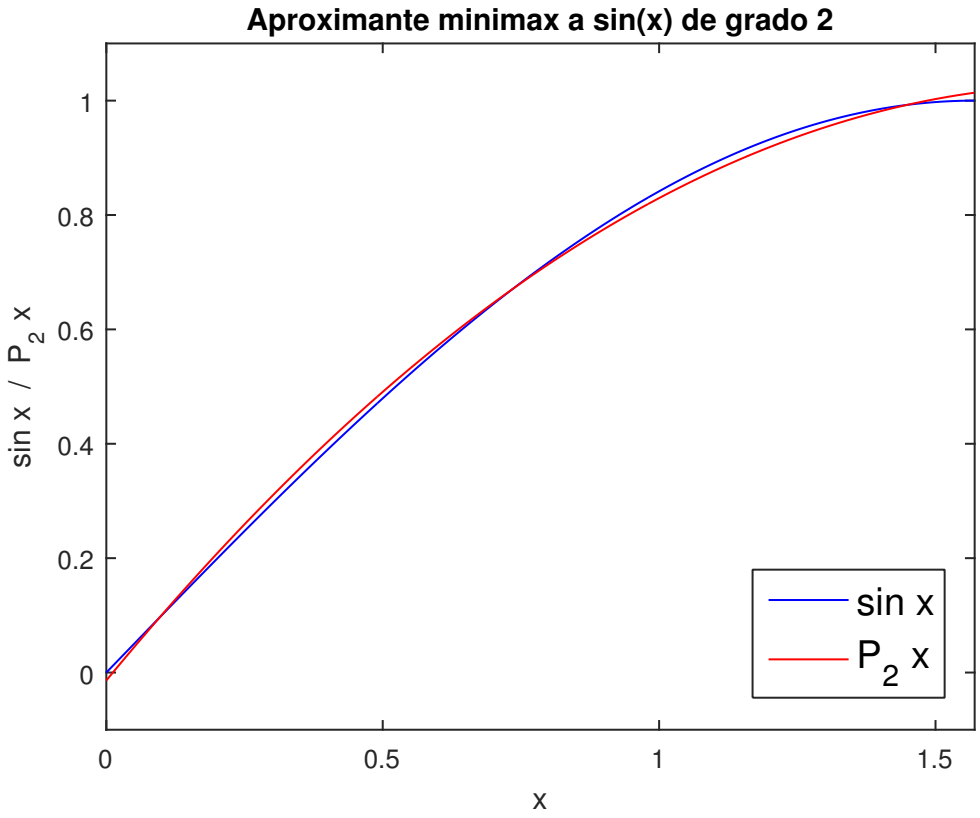

Figura 3.1

3.6.1.2. Ejemplo 2

Consideramos ahora la aproximación minimax por polinomios de grado 5 de la función $f(x) = sin(x)$, en el intervalo $[\pi/4, 3\pi/4]$. La elección del intervalo de aproximación está motivada para que la función a aproximar no sea monótona en dicho intervalo.

El polinomio de aproximación será:

$$B_1+(x-a)\cdot(B_2+(x-a)\cdot(B_3+(x-a)\cdot(B_4+(x-a)\cdot(B_5+(x-a)\cdot B_6))))$$

con

$$B=\begin{vmatrix} 0{,}707116746231351 \\ 0{,}706651699860735 \\ -0{,}350189052216190 \\ -0{,}126915743807098 \\ 0{,}040398535966168 \\ -0{,}000000000000000 \end{vmatrix}$$

La convergencia se alcanza después de cinco iteraciones, y se ilustra en la tabla siguiente, que muestra las sucesivas referencias computadas por el algoritmo.

y_1	y_2	y_3	y_4
0.805089735122384	0.956747316378202	1.230024834963949	1.570796326794897
0.785398163397448	0.908558340961649	1.195411379819388	1.570796326794717
0.785398163397448	0.890966241638338	1.179628735311344	1.570796326795282
0.785398163397448	0.890934960191418	1.178640007112258	1.570796326794944
0.785398163397448	0.890935679822311	1.178640418135171	1.570796326794979
0.785398163397448	0.890935679822204	1.178640418132814	1.570796326794754

Cuadro 3.2: tabla con las referencias y_1 y_2 y_3 y_4

y_5	y_6	y_7
1.911567818625844	2.184845337211591	2.336502918467410
1.946181273770409	2.233034312628262	2.356194490192345
1.961963918278460	2.250626411951306	2.356194490192345
1.962952646477683	2.250657693397972	2.356194490192345
1.962952235455103	2.250656973767282	2.356194490192345
1.962952235457192	2.250656973767686	2.356194490192345

Cuadro 3.3: tabla con las referencias y_5 y_6 y_7

La figura 3.2 muestra el aproximante minimax.

El número de iteraciones necesarias para la convergencia es muy sensible al grado del aproximante que se busca. Con grados altos, no se puede esperar la estabilización de más de cinco cifras decimales significativas de cada punto de la referencia (aunque la tolerancia en el criterio de parada se fije

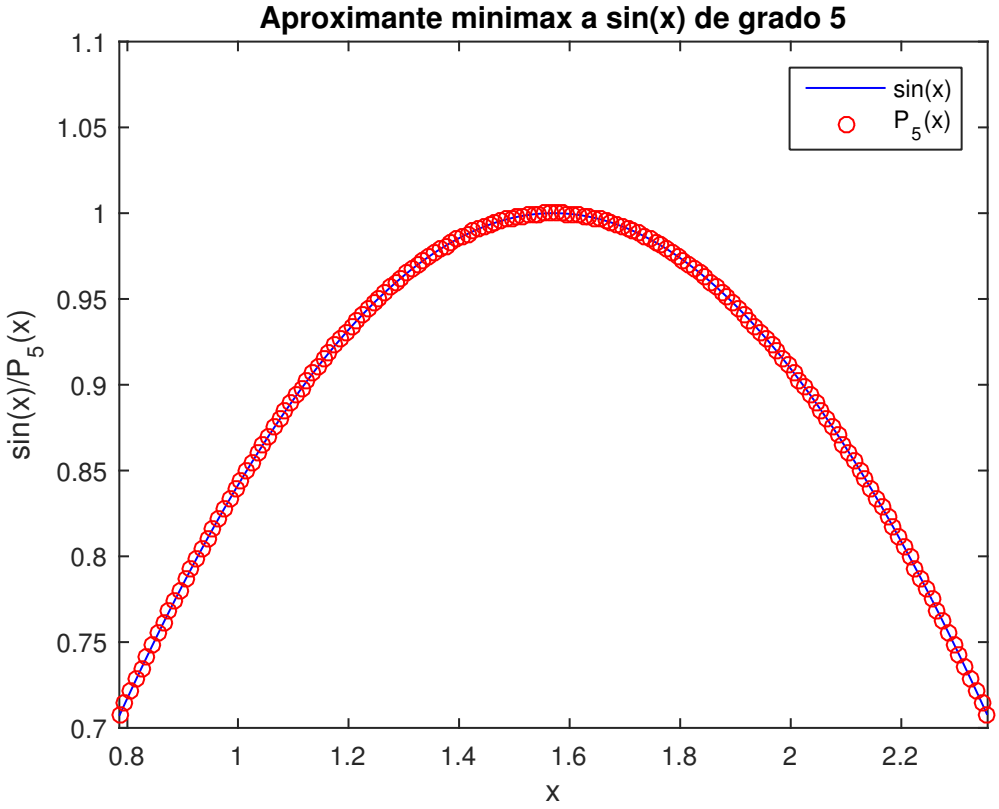

Figura 3.2

en aproximadamente $1{,}0e-9$). Ello causa que la iteración entre en un ciclo infinito, para el que una salvaguarda posible sería fijar también un máximo para el número de iteraciones admisibles (por ejemplo, `maxiter` = 500).

Bibliografía

[1] R. H. Bartels and G. H. Golub, Numerical Analysis: Stable numerical methods for obtaining the Chebyshev solution to an overdetermined system of equations, Association for Computing Machinery Volume 11, Number 6 pp. 401–406 (June, 1968).

[2] E. W. Cheney, Introduction to Approximation Theory, McGraw-Hill Book Company (1966).

[3] A. K. Cline, A Descent Method for the Uniform Solution to Over-Determined Systems of Linear Equations, SIAM J. Numer. Anal. 13(3), pp. 293–309, (1976).

[4] D. Moursund, Chebyshev Solution of $n + 1$ Linear Equations in n Unknowns, Journal of the Association for Computing Machinery, Vol. 12, No. 3, pp. 383 - 387 (July, 1965).

[5] M. J. D. Powell, Approximation theory and methods, Cambridge University Press (1981).

[6] E. Süli & D. Mayers, An Introduction to Numerical Analysis, Cambridge University Press (2003).

[7] N. L. Schryer, Certification of algorithm 328: Chebyshev solution to an overdetermined linear system, by R. H. Bartels and G. H. Golub, Association for Computing Machinery Volume 12, Number 6 pp. 326 (June, 1969).

[8] Tasissa, Abiy, Function approximation and the Remez algorithm (2021).

[9] G. A. Watson, Approximation theory and numerical methods, John Wiley & Sons (1980).