USO DA FATORIZAÇÃO QR NA ESTIMAÇÃO DOS MULTIPLICADORES DE LAGRANGE EM PROGRAMAÇÃO NÃO LINEAR COM RESTRIÇÕES LINEARES

Mischel Carmen Neyra Huamani

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS (M.Sc.) EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO

Aprovada por:

Paulo Roberto Oliveira

Prof. Paulo Roberto Oliveira
(Presidente)

A F Oliveira

Prof. Antonio A. F. de Oliveira

Prof. Álvaro Rodolfo De Pierro

RIO DE JANEIRO, RJ - BRASIL

MARÇO DE 1984

NEYRA HAMANI, MISCHEL CARMEN

Uso da Fatorização QR na Estimação dos Multiplicadores de Lagrange em Programação Não Linear com Restrições Lineares (Rio de Janeiro) 1984.

X, 123 p. 29,7 cm (COPPE/UFRJ, M.Sc., Engenharia de Sistemas e Computação, 1984).

Tese - Universidade Federal do Rio de Janeiro, COPPE.

1. Programação Não Linear I.COPPE/UFRJ II. Título (série).

Aos meus queridos pais:

Justin e Justina

AGRADECIMENTOS

- Ao Professor Paulo Roberto Oliveira pela orientação durante a elaboração deste trabalho.

- Ao Professor Laureano Escudero pela sugestão do tema e orientação recebida.

- Aos Professores Antonio A.F. de Oliveira e Alvaro Rodolfo De Pierro, membros da Banca, pelas sugestões e discussões de grande valia para o trabalho.

- Aos Professores da COPPE - Sistemas pelos ensinamentos transmitidos durante o curso de Pós-Graduação.

- Aos colegas e amigos da COPPE pelo estímulo e companherismo em especial ao amigo Neócles Alves Pereira.

- As amigas Susana Espino de Alayo e Maria Antonieta Amorim dos Santos pelo apoio moral nas horas difíceis.

- A Suely Klajman e Denise Schwartz pela compreensão e paciência demonstradas durante minha passagem pela COPPE.

- Ao CNPq pelo apoio financeiro.

Resumo da Tese Apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

USO DA FATORIZAÇÃO QR NA ESTIMAÇÃO DOS MULTIPLICADORES DE LAGRANGE EM PROGRAMAÇÃO NÃO LINEAR COM RESTRIÇÕES LINEARES

Mischel Carmen Neyra Huamaní

Março de 1984

Orientador: Paulo Roberto Oliveira
Programa: Engenharia de Sistemas e Computação

Diversos algoritmos para problemas de Programação Não Linear Restrita precisam do cálculo de Estimadores dos Multiplicadores de Lagrange, especificamente os algoritmos que seguem a Metodologia dos Conjuntos Ativos.

No presente trabalho descreve-se um algoritmo para o cálculo desses estimadores no caso de Programação Não Linear Restrita Linearmente.

Apresenta-se os fundamentos da fatorização QR e suas atualizações, após uma adição ou apagamento de restrição (linha) ou variável (coluna) ter sido feita na matriz corrente, assim como também dois procedimentos para obter a matriz Q.

Abstract of Thesis presented to COPPE/UFRJ as partial full-filment of the requirements for the degree of Master of Science (M.Sc.)

QR FACTORIZATION IN THE COMPUTATION OF LAGRANGE MULTIPLIERS ESTIMATES IN LINEARLY CONSTRAINED NON LINEAR PROGRAMMING

Mischel Carmen Neyra Huamaní

March, 1984

Chairman: Paulo Roberto Oliveira
Department: Engenharia de Sistemas e Computação

Several algorithms for Constrained Nonlinear Programming require the computation of Lagrange Multipliers Estimates, specifically the algorithms following the Active Set Methodology.

In this work we discuss an algorithm for computing these estimates in Linearly Constrained Nonlinear Programming.

We discuss the foundations of QR factorization and its updatings after a constrained (row) or variavel (column) has been added or deleted in the current matrix. We also presented two procedures in order to obtain the Q matrix.

ÍNDICE

Páginas

CAPÍTULO I - INTRODUÇÃO ... 1

CAPÍTULO II - PROBLEMA DE MÍNIMOS QUADRADOS ... 3

2.1. Motivação ... 3

2.2. Enunciado do Problema ... 3

CAPÍTULO III - TRANSFORMAÇÕES ORTOGONAIS ... 9

3.1. Transformações de Householder ... 9

3.2. Transformações de Givens ... 15

CAPÍTULO IV - TRIANGULARIZAÇÃO DE MATRIZES POR TRANSFORMAÇÕES DE GIVENS ... 21

4.1. Redução de um vetor v(txl) a um múltiplo de e_1 ... 21

4.2. Triangularização de uma matriz quadrada ... 22

4.3. Triangularização de uma matriz de Hessenberg superior ... 25

4.4. Triangularização de uma matriz triangular superior com um vetor A_p anexado ... 27

4.4.1. A_p vetor linha na última posição ... 27

4.4.2. A_p vetor linha em posição intermediária ... 28

4.4.2.1. Uso de Transformações de Givens ... 29

4.4.3. A_p vetor coluna em posição intermediária ... 30

Páginas

CAPÍTULO V - FATORIZAÇÃO QR 34

5.1. Definição 34

5.2. Fatorização QR segundo as dimensões de A 34

5.2.1. Caso de sistema sobredeterminado (n > t) 34

5.2.1.1. Posto (A) = k < t 36

5.2.2. Caso de sistema indeterminado (n < t) 37

5.3. Fatorização QR para resolver o Problema dos Mínimos Quadrados 38

CAPÍTULO VI - FATORIZAÇÃO QR PARA SISTEMAS SOBREDETERMINADOS 45

6.1. Cálculo das matrizes Q e R 45

6.2. Procedimentos para obter a matriz Q 46

6.2.1. Algoritmo Q Normalizado (NQA). 47

6.2.2. Algoritmo Q Não Normalizado (NNQA) 50

6.3. Cálculo do vetor residual r = b - Ax. 52

6.4. Cálculo do vetor de incógnitas . . . 53

CAPÍTULO VII - ATUALIZAÇÕES DA FATORIZAÇÃO QR 57

7.1. Modificação de uma matriz ortogonal . 59

7.2. Modificação de Posto Um 62

7.3. Adição de um vetor coluna à matriz A. 65

7.4. Apagar um vetor coluna da matriz A . 71

7.5. Adição de um vetor linha à matriz A . 76

7.6. Apagar um vetor linha da matriz A . . 79

Páginas

CAPÍTULO VIII - CÁLCULO DOS ESTIMADORES DOS MULTIPLICADORES DE LAGRANGE PARA PROBLEMAS NÃO LINEARES RESTRITOS LINEARMENTE ... 92

8.1. Motivação: Estratégia dos Conjuntos Ativos em PNL com Restrições Lineares ... 92

8.1.1. Enunciado do Problema ... 92

8.1.2. Condições de Otimalidade para o Problema P1 ... 93

8.1.3. Metodologia da Estratégia dos Conjuntos Ativos ... 94

8.1.4. Importância do sinal dos estimadores ... 95

8.1.5. Erros no cálculo dos estimadores dos Multiplicadores de Lagrange (M-L) ... 97

8.1.6. Aplicações dos estimadores dos M-L ... 97

8.2. Cálculo dos estimadores dos M-L no Problema Quadrático ... 98

8.2.1. Enunciado do Problema ... 98

8.2.2. Condições de Otimalidade para o Problema Quadrático ... 99

8.2.2.1. Cálculo da direção viável ... 100

8.2.3. Aplicação da Estratégia de Conjuntos Ativos no Problema Quadrático ... 100

8.2.4. Obtenção dos estimadores dos M-L no Problema Quadrático ... 101

Páginas

8.2.4.1. Pelas Equações Normais 102

8.2.4.2. Pela Fatorização QR. 102

8.2.4.2.1. Estimadores de Primeira Ordem dos M-L no Problema Quadrático 103

8.2.4.3. Usando um ponto arbitrário 105

8.2.4.4. Usando o valor de QR 105

8.2.5. Conclusões 106

8.3. Cálculo dos estimadores dos M-L para um problema geral de PNL com Restrições Lineares 107

8.3.1. Enunciado do Problema 107

8.3.2. Condições de Otimalidade . . 107

8.3.2.1. Cálculo da direção viável 108

8.3.3. Aplicação da Estratégia de Conjuntos Ativos ao Problema Não Linear com Restrições Lineares 108

8.3.4. Cálculo dos estimadores de Primeira Ordem dos M-L . . . 109

8.3.5. Cálculo dos estimadores de Segunda Ordem dos M-L 110

8.3.5.1. Alternativa I : Estimando d 111

Páginas

8.5.3.2. Alternativa II : Usando QR 111

8.3.5.2.1. Cálculo do estimador δ 112

8.5.3.3. Conclusões . . . 113

8.3.6. Cálculo dos estimadores Pseudo Segunda Ordem dos M-L 113

8.4. Análise dos estimadores 114

8.5. Conclusões 116

CAPÍTULO IX - CONCLUSÕES GERAIS 118

BIBLIOGRAFIA 120

CAPÍTULO I

INTRODUÇÃO

Nos problemas de Programação Não Linear com Restrições Lineares, existem diversos métodos para encontrar uma solução, que, no melhor dos casos, será ótima.

Nestes métodos, os Multiplicadores de Lagrange (M-L) são importantes porque permitem testar uma das condições de otimalidade. Mas, nos métodos baseados na Estratégia de Conjuntos Ativos, os M-L tem maior importância na convergência do algoritmo.

Geralmente não é possível calcular exatamente esses M-L, mas sim obter os estimadores deles, gerando um problema de Mínimos Quadrados que pode ser resolvido usando fatorização QR. Mas ainda, nos métodos de conjuntos ativos, em cada iteração modifica-se o Conjunto Ativo seja adicionando ou apagando uma restrição ou uma variável. Nesse caso não é preciso calcular novamente os fatores QR, só se deve atualizar a matriz obtida anteriormente.

A presente dissertação tem por objetivo apresentar os resultados de pesquisas sobre os tópicos citados, que se encontram disseminados em diversos artigos e livros tentando-se aqui fornecer uma visão geral deles, o qual, acreditamos, ainda não está publicado num único texto.

Portanto, no Capítulo II, expõe-se o Problema dos Mínimos Quadrados, que será resolvido usando fatorização QR baseado nas propriedades das Transformações Ortogonais.

Nos Capítulos III e IV são apresentados os aspectos teóricos e aplicações das Transformações Ortogonais.

Nos Capítulos V e VI, apresentam-se os fundamentos da fatorização QR e procedimentos desenvolvidos recentemente para obter a matriz Q.

No Capítulo VII detalha-se as atualizações da fatorização QR, e finalmente no Capítulo VIII apresenta-se uma aplicação na obtenção dos estimadores dos M-L para problemas Não-Lineares restritos linearmente.

CAPÍTULO II

PROBLEMA DE MÍNIMOS QUADRADOS LINEARES

2.1. Motivação

O problema de Mínimos Quadrados Lineares é apresentado num capítulo isolado dada sua importância e aplicação nas diversas áreas.

Geralmente para qualquer problema com dados que sobredeterminem uma solução, recorre-se a um Método de Aproximação sendo o de Mínimos Quadrados o mais usado.

No caso linear tem-se x como o vetor de incógnitas no problema de ajuste de uma função linear Ax a um vetor de dados reais denotado por b .

Em Programação Matemática, tem-se que A é a matriz de restrições e b o vetor a ser satisfeito. Se o sistema $Ax = b$ não é compatível, o objetivo é obter um ponto x_L tal que $||b - Ax||_2^2$ seja mínimo.

Uma outra aplicação temos em Otimização restrita não Linear quando se vai obter os Multiplicadores de Lagrange associados às restrições ativas. Uma explicação mais detalhada será dada no capítulo VIII.

2.2. Enunciado do Problema

Dada uma matriz $A(nxt)$ de posto $k \leq \min(n,t)$ e dado um vetor $b(nx1)$, encontrar um vetor $x(tx1)$ tal que minimize a norma euclideana do vetor residual r , onde:

$$||r||^2 = ||b - Ax||^2 \tag{2.1}$$

Em algumas aplicações é possível usar a norma um ou a norma infinito, mas a solução conduz a um problema combinatório difícil.

Demonstram-se as condições necessárias e suficientes para que x seja solução do problema de Mínimos Quadrados.

TEOREMA 1:

A condição necessária e suficiente para que x_L seja solução dos Mínimos Quadrados é:

$$A^T(b - Ax_L) = 0 \tag{2.2}$$

onde A^T é a transposta da matriz A. O ponto x_L é único se A tiver posto completo $(k = \text{mín}(n,t))$.

a) <u>Necessidade</u>

Suponha que a x corresponde o resíduo $||r||^2 = ||b\text{-}Ax||^2$ mínimo, mas:

$$A^T(b - Ax) = s \neq 0$$

O resíduo correspondente a $(x + \eta s)$ onde η é um escalar é:

$$\begin{aligned}||b - A(x + \eta s)||^2 &= ||b - Ax - A\eta s||^2 \\ &= ||b - Ax||^2 + \eta^2 ||As||^2 - 2\eta s^T A^T (b-Ax) \\ &\leq ||b - Ax||^2 + ||A||^2 ||s||^2 \eta^2 - 2||s||^2 \eta \\ &\leq ||b - Ax||^2 + ||s||^2 \eta \left[||A||^2 \eta - 2\right] \\ &< ||b - Ax||^2\end{aligned}$$

para $0 < \eta < \dfrac{2}{||A||^2}$ o que contradiz a hipótese de resíduo mínimo.

b) Suficiência

Suponha $A^T(b - Ax_L) = 0$. Seja $r(\varepsilon)$ o vetor residual correspondendo a outro vetor qualquer $(x_L + \varepsilon)$, onde ε é também um vetor, então:

$$||r(\varepsilon)||^2 = ||b - A(x_L + \varepsilon)||^2$$
$$||r(\varepsilon)||^2 = ||b - Ax_L - A\varepsilon||^2$$
$$||r(\varepsilon)||^2 = ||b - Ax_L||^2 + ||A\varepsilon||^2 - 2(b - Ax_L)^T A\varepsilon$$
$$||r(\varepsilon)||^2 = ||b - Ax_L||^2 + ||A\varepsilon||^2$$
$$||r(\varepsilon)||^2 \geq ||b - Ax_L||^2$$

c) Unicidade

Em b), se $||A\varepsilon|| > 0$, para todo ε, é evidente que a solução é única. Admita portanto que $A\bar{\varepsilon} = 0$ para algum vetor $\bar{\varepsilon}$, o que tornaria $x_L + \bar{\varepsilon}$ solução do problema dos Mínimos Quadrados.

Entretanto, isto só é possível com $\bar{\varepsilon} = 0$, dado ser A de posto completo.

Da hipótese do Teorema 1, $A^T(b - Ax) = 0$, temos o seguinte sistema de equações chamadas Equações Normais:

$$A^T Ax = A^T b \tag{2.3}$$

O sistema anterior terá uma solução se a matriz $(A^T A)$ for não singular; condição para isto, será demonstrada no seguinte teorema:

TEOREMA 2

A matriz $A^T A$ é não singular se e somente se as colunas de A são linearmente independentes (posto coluna completo). A demonstração encontra-se em DAHLQUIST-BJÖRCK (2).

Portanto, quando as colunas de A são linearmente independentes, a solução x_L é única:

$$x_L = A^+ b \tag{2.4}$$

onde A^+ é a Pseuda-Inversa da matriz A. Existem diversas definições para A^+, mas a mais utilizada é:

$$A^+ = (A^T A)^{-1} A^T \tag{2.5}$$

assumindo que $(A^T A)$ é não singular, desde que A tem posto completo.

Se A for mal condicionada, a matriz $A^T A$ também é, evitando-se usar as equações normais.

Portanto, utilizam-se os Métodos de Ortogonalização como alternativa para encontrar a solução ao problema de Mínimos Quadrados.

Devido à invariância da norma euclideana na pré-multiplicação de vetores por matrizes ortogonais temos para qualquer vetor y(nx1) e qualquer matriz ortogonal Q(nxn) que:

$$||Qy|| = ||y||$$

Basta ver que $||Qy||^2 = y^T Q^T Qy = y^T y = ||y||^2$.

No caso do problema de Mínimos Quadrados que consiste em minimizar a norma euclideana do vetor residual, temos:

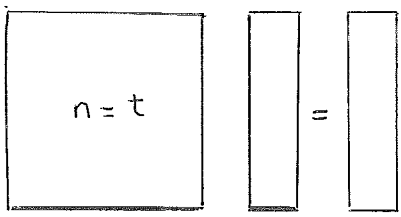
$$||Q(Ax - b)|| = ||QAx - Qb|| = ||Ax - b||$$

para qualquer matriz ortogonal Q(n,n) e qualquer vetor x(tx1).

As transformações ortogonais mais usadas são: transformações de Householder e de Givens que serão apresentadas no seguinte capítulo.

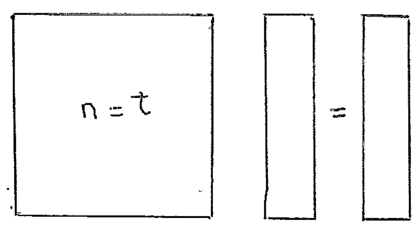
Dependendo das dimensões relativas de n,t e posto (A) é conveniente classificar os diferentes casos em:

Caso 1a: (n=t)
Determinado (posto completo)



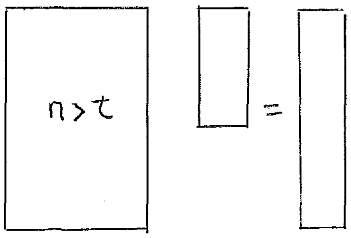
Ax = b
posto(A)=n=t

Caso 1b: (n=t)
Sem solução ou indeterminado (posto deficiente)



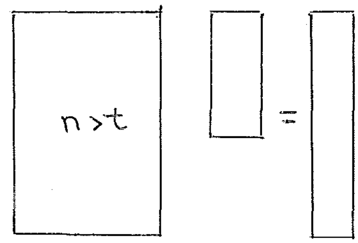
Ax≅b
posto(A)=k < n = t

<u>Caso 2a</u>: $(n > t)$
Sobredeterminado (ou determinado) posto completo



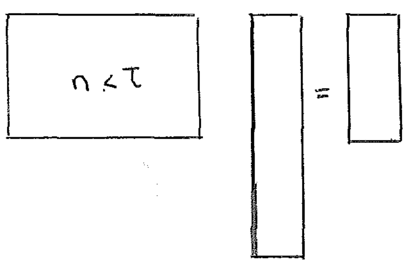
$Ax \cong b$ (ou $Ax=b$)
$\text{posto}(A) = t < n$

<u>Caso 2b</u>: $(n > t)$
Sobredeterminado
posto deficiente



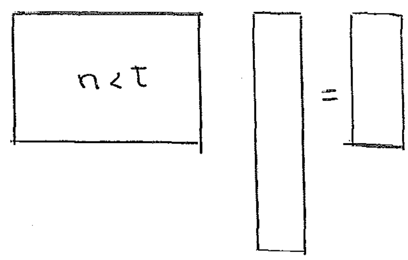
$Ax \cong b$
$\text{posto}(A)=k < t < n$

<u>Caso 3a</u>: $(n < t)$
Indeterminado
posto completo



$Ax=b$
$\text{posto}(A) = n < t$

<u>Caso 3b</u>: $(n < t)$
Indeterminado
posto deficiente



$Ax \cong b$
$\text{posto}(A)=k < n < t$

Os três casos apresentados, serão tratados no capítulo V ao expor a fatorização QR para cada um deles.

CAPÍTULO III

TRANSFORMAÇÕES ORTOGONAIS

No capítulo anterior, mencionou-se os Métodos de Ortogonalização como alternativa à formação das equações normais para a resolução do Problema de Mínimos Quadrados.

Estes métodos são convenientes devido a sua estabilidade nos cálculos já que as normas de vetores são invariantes por pré-multiplicação por matrizes ortogonais.

3.1. Transformações de Householder

A transformação de Householder é definida a partir de uma matriz elementar simétrica da forma:

$$H = I - \frac{1}{\beta} w\, w^T \tag{3.1}$$

onde $\beta = ||w||_2^2 / 2$, sendo w um vetor qualquer não nulo.

Uma matriz de Householder é ortonormal:

Com efeito,

$$H^T H = (I - \frac{1}{\beta} w\, w^T)^T (I - \frac{1}{\beta} w\, w^T)$$

$$H^T H = I - \frac{2}{\beta} w\, w^T + \frac{1}{\beta^2} w\, w^T . w\, w^T = I$$

Sejam agora dois vetores quaisquer e diferentes

a e b de igual comprimento Euclideano; existe uma matriz de Householder que transforma a em b , isto é:

$$Ha = b$$

Aplicando a transformação, temos:

$$Ha = (I - (ww^T)/\beta)a$$
$$Ha = a - w(w^T a/\beta) \tag{3.2}$$
$$Ha = b$$

De (3.2) concluimos que o vetor transformado Ha é simplesmente uma diferença entre o vetor original a e um múltiplo do vetor Householder w . Portanto, o vetor transformado é idêntico ao original em todas suas componentes onde o vetor w de Householder é zero.

A transformação de Householder e as Matrizes de Givens, permitem reduzir um vetor coluna dado, digamos v,num múltiplo de uma coluna da matriz identidade. Isso significa encontrar uma matriz ortonormal $P(txt)$ tal que:

$$Pv = \pm \rho e_t \qquad \text{onde } \rho \text{ é o múltiplo} \tag{3.3}$$

No caso da transformação de Householder veremos que uma determinada sequência delas, seja $H_1 H_2 \ldots H_p = Q$, transforma uma matriz A numa matriz triangular superior R ie:

$$Q^T A = \begin{bmatrix} R \\ 0 \end{bmatrix} \tag{3.4}$$

Seja H_ℓ uma transformação de Householder da seguinte forma:

$$H_\ell = I - (w_\ell w_\ell^T)/\beta_\ell \tag{3.5}$$

onde as primeiras ℓ-1 componentes de w_ℓ são zeros. Cada H_ℓ é escolhida para introduzir zeros nas últimas n-ℓ componentes da ℓ-ésima coluna de A.

Vejamos um exemplo para entender o processo da redução. Seja A uma matriz (nxt) = (5x3), de posto completo

$$A = \begin{bmatrix} x & x & x \\ x & x & x \\ x & x & x \\ x & x & x \\ x & x & x \end{bmatrix} \tag{3.6}$$

A transformação de Householder H_1, é tal que o produto de H_1 e a primeira coluna de A é zero nas últimas (n-1) = 4 componentes.

Ao aplicar esta transformação à matriz A, introduzem zeros nos elementos com círculos em (3.6), tendo a matriz H_1A a seguinte forma:

$$H_1A = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ 0 & x & x \\ 0 & x & x \\ 0 & x & x \\ 0 & x & x \end{bmatrix} \tag{3.7}$$

Agora, uma matriz H_2 é escolhida para introduzir zeros nos elementos com círculos em (3.7). Acontece que a primeira componente do vetor w_2 que determina H_2 é zero e então os componentes já zerados e os chamados r_{ij} são inalteráveis pela aplicação de H_2 (veja (3.5)) a H_1A, obtendo-se o seguinte:

$$H_2H_1A = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ 0 & r_{22} & r_{23} \\ 0 & 0 & x \\ 0 & 0 & x \\ 0 & 0 & x \end{bmatrix} \tag{3.8}$$

Por último a matriz de transformação H_3 é escolhida para introduzir zeros nos elementos com círculos em (3.8), ficando novamente inalteráveis os elementos já zerados e os r_{ij}, obtendo-se finalmente:

$$H_3H_2H_1A = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ 0 & r_{22} & r_{23} \\ 0 & 0 & r_{33} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{3.9}$$

Observando a expressão (3.4) e (3.9), elas têm a mesma forma. Por outro lado, desde que cada matriz H_ℓ é ortogonal, o produto $H_3\ H_2\ H_1$ também o é.

Construção das matrizes de Householder:

O primeiro passo da redução será desenvolvido teoricamente, sendo os demais similares.

Seja a_1 a primeira coluna da matriz A . Escolhe-se w_1 e β_1 em (3.5) tal que:

$$H_1 \, a_1 = \rho \, e_1 \tag{3.10}$$

com H_1 ortogonal e $\rho = \pm \, ||a_1||$ onde o sinal será escolhido de modo a manter o sinal da primeira coordenada de a_1, conforme (3.6). O valor absoluto de ρ já leva em conta a propriedade de H de invariância da norma euclideana do vetor transformado e isso será demonstrado.

Define-se:

$$w_1 = e_1 + a_1/\sigma||a_1|| \tag{3.11}$$

onde $\sigma = \pm 1$ segundo a primeira componente de a_1 . As primeiras componentes de w_1 e a_1 serão respectivamente β_1 e ε onde:

$$\beta_1 = 1 + \varepsilon/\sigma||a_1|| \tag{3.12}$$

Usando a definição (3.5) para H_1 , tem-se:

$$H_1 \, a_1 = (I - (w_1 w_1^T)/\beta_1) a_1$$

$$H_1 \, a_1 = a_1 - \left[(e_1+a_1/\sigma||a_1||)(e_1+a_1/\sigma||a_1||)^T \, a_1\right]/(1+\varepsilon/\sigma||a_1||)$$

$H_1\ a_1 = a_1 - |(e_1 + a_1/\sigma||a_1||)(\varepsilon + \sigma||a_1||)|/(1+\varepsilon/\sigma||a_1||)$

$H_1\ a_1 = a_1 - [(\sigma||a_1||e_1 + a_1)(\varepsilon + \sigma||a_1||)]/(\sigma||a_1|| + \varepsilon)$

$H_1\ a_1 = -\sigma||a_1||\ e_1$

$$H_1\ a_1 = \rho\ e_1 \qquad (3.13)$$

onde $\rho = -\sigma\ ||a_1||$

Aplicando a transformação H_1 às colunas da matriz A , obtemos:

$A = (a_1,\ a_2, \ldots\ a_t)$

$H_1\ A = (H_1 a_1,\ H_1 a_2,\ \ldots\ H_1 a_t)$

O cálculo de $H_1 a_1$ já foi amostrado, obtendo-se uma coluna com todas suas componentes nulas exceto a primeira.

$$r_{11} = -\sigma||a_1|| \qquad (3.14)$$

Para calcular $H_1\ a_j$ para $j > 1$, temos:

$$H_1\ a_j = (I - (w_1 w_1^T)/\beta_1)a_j \qquad (3.15)$$

$$H_1\ a_j = a_j - ((w_1^T\ a_j)/\beta_1)w_1$$

Tem-se uma sequência de cálculos:

$$z = (w_1^T\ a_j)/\beta_1 \qquad (3.16)$$

$$a_j = a_j - z\, w_1 \qquad (3.17)$$

com a qual a Transformação de Householder H_1 é determinada para o vetor w_1.

3.2. Transformações de Givens

As transformações de Givens, mais simples que as de Householder, são Rotações de Planos usadas principalmente para anular um elemento de um vetor.

Estas transformações são equivalentes às de Householder, onde o vetor de Householder tem elementos diferentes de zero só nas posições i e j:

Uma matriz de rotação de plano, diferencia-se da matriz identidade só nas linhas e colunas i e j.

Assumindo que $i < j$, então uma rotação do plano (i,j) é representada por uma matriz da forma:

$$P_j^i = \begin{array}{c} \\ i \\ \\ j \\ \\ \end{array} \begin{array}{c} \begin{array}{ccccc} & i & & j & \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & c & 0 & s & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & s & 0 & -c & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array} \qquad (3.18)$$

onde $c = \cos\theta$, $s = \operatorname{sen}\theta$ para algum θ e $c^2 + s^2 = 1$.

Por outro lado, tem-se:

$$\begin{bmatrix} c & s \\ s & -c \end{bmatrix} \begin{bmatrix} c & s \\ s & -c \end{bmatrix} = I \qquad (3.19)$$

e então as matrizes P^i_j são ortonormais i.e.:

$$P^i_j \, P^{i^T}_j = I \quad e \quad P^{i^T}_j \, P^i_j = I \tag{3.20}$$

DONGARRA et alli (4) enfatizam que as transformações de Givens possuem duas propriedades interessantes:

1) Uma rotação do plano requer pouco trabalho ao ser aplicada num vetor.

Seja $\bar{v} = P^i_j \, v$

$$\bar{v} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & c & 0 & s & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & s & 0 & -c & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{bmatrix} = \begin{bmatrix} v_1 \\ cv_2 + sv_4 \\ v_3 \\ sv_2 - cv_4 \\ v_5 \end{bmatrix}$$

Em geral:

$$\begin{aligned} \bar{v}_i &= cv_i + sv_j \\ \bar{v}_j &= sv_i - cv_j \\ \bar{v}_k &= v_k \qquad \text{para} \quad k \neq i \; , \; k \neq j \end{aligned} \tag{3.21}$$

2) A matriz P^i_j pode ser escolhida para anular um elemento do vetor, seja i ou j .

2.a) Anulando $\overline{v}_j$:

Seja $\rho^2 = v_i^2 + v_j^2$, e se quer anular $\overline{v}_j$. Considere a mesma matriz P_j^i , obtendo:

$$\overline{v}_j = sv_i - cv_j = 0$$

donde $sv_i = cv_j$ e $s = v_j/\rho$, $c = v_i/\rho$. Neste caso $\overline{v}_i$ será calculado usando os valores de c e s , i.e.:

$$\overline{v}_i = cv_i + sv_j$$

$$\overline{v}_i = c^2\rho + s^2\rho$$

$$\overline{v}_i = \rho \qquad (3.22)$$

No exemplo acima teríamos:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & c & 0 & s & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & s & 0 & -c & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{bmatrix} = \begin{bmatrix} v_1 \\ \rho \\ v_3 \\ 0 \\ v_5 \end{bmatrix}$$

onde:

$$\rho = v_2^2 + v_4^2 , \quad c = v_2/\rho , \quad s = v_4/\rho$$

2.b) Anulando $\bar{v}_i$:

Neste caso, usa-se a matriz P_i^j com os elementos:

$$P_i^j(i,i) = -c \qquad P_i^j(j,j) = c$$
$$P_i^j(i,j) = P_i^j(j,i) = s \tag{3.23}$$

Seja $\bar{v} = P_i^j \, v_j$:

$$\bar{v}_i = -cv_i + sv_j$$
$$\bar{v}_j = sv_i + cv_j$$
$$\bar{v}_k = v_k \qquad \text{para} \quad k \neq i \; , \; k \neq j$$

Anulando $\bar{v}_i$: $sv_j = cv_i$ e $s = v_i/\rho$, $c = v_j/\rho$ onde o valor de $\bar{v}_j$ será calculado usando os valores de c e s i.e.:

$$\bar{v}_j = sv_i + cv_j$$
$$\bar{v}_j = s^2\rho + c^2\rho$$
$$\bar{v}_j = \rho \tag{3.24}$$

No exemplo teríamos:

$$\bar{v}=P_2^4 v_4 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & -c & 0 & s & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & s & 0 & c & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{bmatrix} = \begin{bmatrix} v_1 \\ 0 \\ v_3 \\ \rho \\ v_5 \end{bmatrix}$$

onde:

$$\rho^2 = v_2^2 + v_4^2 \ , \ c = v_4/\rho \ , \ s = v_2/\rho$$

Assim uma sequência de rotações de plano pode ser escolhida para introduzir zeros dentro de uma matriz; mas não arbitrariamente, desde que rotações posteriores podem destruir zeros já introduzidos.

Esta sequência de rotações de plano, é escolhida para zerar elementos num vetor ou uma matriz. Se por exemplo se transforma o vetor $v = (v_1, v_2, \ldots, v_i, \ldots v_n)^T$ no vetor $\bar{v} = \bar{v}_i\, e_i$ usa-se:

$$P_1^i \ldots P_{i-1}^i \ P_{i+1}^i \ \ldots P_n^i \ v = \bar{v}_i e_i \qquad (3.25)$$

onde e_i é o i-ésimo vetor unitário (canônico) e <u>P_i^j é a transformação de Givens que anula o elemento i na coluna j.</u> Desde que P_j^i é ortogonal temos $v^t v = \bar{v}_i^2$.

Portanto, usando transformações de Givens, uma matriz $A(nxt)$ de posto completo pode ser decomposta segundo:

$$QA = \begin{bmatrix} R \\ 0 \end{bmatrix}$$

onde:

$$Q = P_t \ P_{t-1} \ \ldots \ P_1 \qquad (3.26)$$

A expressão P_1 , por exemplo, é na realidade um produto de matrizes de Givens:

$$P_1 = P_2^1 \, P_3^1 \ldots P_{n-1}^1 \, P_n^1$$

Tal que P_1A tem como primeira coluna um múltiplo de e_1 .Em seguida, aplicando P_2 à matriz P_1A , onde:

$$P_2 = P_3^2 \, P_4^2 \ldots P_{n-1}^2 \, P_n^2$$

obtem-se $P_2 \, P_1 \, A$ cuja segunda coluna é zerada do elemento 3 até n , e os elementos já zerados das primeiras colunas permanecem inalterados ; segundo esse procedimento obtemos finalmente a matriz Q , segundo a expressão (3.26).

CAPÍTULO IV

TRIANGULARIZAÇÃO DE MATRIZES POR TRANSFORMAÇÕES DE GIVENS

As transformações a serem apresentadas neste capítulo serão usadas posteriormente quando sejam estudadas as atualizações da fatorização QR.

4.1. Redução de um vetor $v(t \times 1)$ a um múltiplo de e_1

Seja $Pv = \rho e_1$ onde $\rho = ||v||_2$, então P pode ser escrito como:

$$P = P_2^1 \, P_3^1 \ldots P_t^1 \tag{4.1}$$

Da mesma forma é possível reduzir um vetor linha $v^T(1 \times t)$ a um múltiplo de e_1^T tal que:

$$v^T \, P^T = \rho \, e_1^T$$

será obtido aplicando P^T ao vetor v^T onde:

$$P^T = P_t^1 \ldots P_3^1 \, P_2^1 \tag{4.2}$$

já que $P_j^{iT} = P_j^i$.

4.2. Triangularização de uma matriz quadrada

O objetivo é obter uma matriz triangular R tal que: $PA = R$

onde P representa uma sucessão de transformações de Givens. Observe que, em consequência da ortogonalidade de P, obtém-se, da identidade:

$$A^T P^T P A = R^T R ,$$

que

$$A^T A = R^T R$$

uma decomposição que pode ser útil na resolução de equações normais. A decomposição citada coincide com a Decomposição de Choleski, se $(A^T A)$ fosse simétrica e definida positiva.

A seguir exemplificaremos com uma matriz (3x3), passando em seguida ao caso geral.

Passo 1: o vetor coluna A_1 da matriz A é transformado no vetor coluna $R_1 = (r_{11}, 0, 0)^T$ de R :

$$P_2^1 P_3^1 A = (R_1, \overline{A}_2, \overline{A}_3) \equiv \overline{A}$$

A rotação do plano P_j^1 para $j = 2,3$ é obtida modificando a matriz identidade I com os escalares correspondentes c e s.

Deve ter-se cuidado que os escalares c e s da matriz de rotação de plano P_2^1 sejam obtidos dos elementos 1 e 2 do vetor coluna $P_3^1 A_2$ de modo que $\overline{A}_2 = P_2^1 P_3^1 A_2$ e $\overline{A}_3 = P_2^1 P_3^1 A_3$.

Passo 2 : O vetor coluna A_2 da matriz $\bar{A}$ é transformado no vetor coluna $R_2 = (r_{12}, r_{22}, 0)^T$ de R ie :

$$P_3^2 \bar{A} = (R_1, R_2, \bar{\bar{A}}_3) = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ & r_{22} & r_{23} \\ & & r_{33} \end{bmatrix} \qquad (4.3)$$

A aplicação da matriz P_3^2 no plano (2,3) ao vetor R_1 , não produz mudança nenhuma nesse vetor.

Desde que P_j^i v mantém o elemento v_k inalterado para $k \neq i$, $k \neq j$ e sendo R uma matriz triangular superior temos, para $j > i$ em P_j^i , $r_{\ell h} = 0$ para $\ell > h$ .Sendo assim, a matriz de rotação de plano P_j^i no plano (i,j) não afeta o vetor R_h para $h < i$.

PROCEDIMENTO GERAL:

Seja A uma matriz (nxn)

Passo 1 : Faça $k = 0$ e $\bar{A} = A$

Passo 2 : Se $k = n-1$ então $\bar{A}$ é a matriz triangular superior R .

Passo 3 : Faça $k = k+1$ e $\bar{A} = P_{k+1}^k \dots P_n^k A$
Vá ao passo 2

OBSERVAÇÕES:

Do que vimos acima podemos concluir:

(1) Para obter as matrizes de Givens P_j^k nos planos (k,j) para $j = n,\dots k+1$ em cada passo k, os escalares correspondentes c e s são calculados usando os elementos j e k do vetor coluna $\bar{A}_k$ da matriz $\bar{A}$ que foi obtida na última rotação.

(2) A última rotação pode ser realizada no mesmo passo ou no anterior mas só para $j = k+1$.

(3) Fazendo $\bar{A} = P^k \bar{A}$ no passo 3, só realiza a operação $P_j^k \bar{A}_i$ para $j = n,\dots k+1$ e $i = k\dots n$.

(4) A operação $P_j^k \bar{A}$ só modifica o elemento $\bar{a}_{kk}$ tal que $\bar{a}_{kk} = (\bar{a}_{kk}{}^2 + \bar{a}_{kj}{}^2)^{1/2}$ e $\bar{a}_{kj} = 0$.

Resumindo o procedimento e observações anteriores, temos o seguinte algoritmo para obter a matriz triangular superior R.

ALGORITMO:

Passo 1 : Faça $k = 0$.

Passo 2 : Se $k = n-1$ então A é a matriz triangular superior R.

Passo 3 : Faça $k = k+1$

Obter $P_j^k A$ para $j = n,\dots k+1$

Obter $P_j^k A_k$:

$a = (a_{kk}^2 + a_{jk}^2)^{1/2}$

$c = a_{kk}/a$

$s = a_{jk}/a$

$a_{kk} = a$

$a_{jk} = 0$

Obter $P_j^k A_i$ para $i = k+1,\dots,n$

$a = ca_{ki} + sa_{ji}$

$b = -sa_{ki} + ca_{ji}$

$a_{ki} = a$

$a_{ji} = b$

Vá ao passo 2.

4.3. Triangularização de uma matriz de Hessenberg superior

Uma matriz A é dita a ser Hessenberg superior se $a_{ij} = 0$ $(i \geq j+2)$ e Hessenberg inferior se $a_{ij} = 0$ $(j \geq i+2)$. A triangularização de uma matriz Hessenberg, é semelhante no caso quadrado.

Para explicar o algoritmo de triangularização, consideremos $\hat{R}$ uma matriz de Hessenberg superior (t,t-1):

$$\hat{R} = \begin{bmatrix} x & x & x & x & x \\ & x & x & x & x \\ & & x & x & x \\ & & x & x & x \\ & & & x & x \\ & & & & x \end{bmatrix}$$

Notemos por p o número da coluna em que aparece o primeiro elemento não-nulo fora da parte triangular de $\hat{R}$ (na figura p=3).

No exemplo, a matriz apresentada, corresponde a uma matriz triangular superior onde foi apagada a terceira coluna.

Aplicando transformações de Givens, na sequência indicada em (4.5), à matriz $\hat{R}$; obtemos uma matriz triangular superior $\binom{\bar{R}}{0}$ onde $\bar{R}$ é uma matriz $((t-1)x(t-1))$ e 0 é o vetor zero (t-1):

$$P_{t}^{t-1} \ldots P_{p+2}^{p+1} \; P_{p+1}^{p} \; \hat{R} = \binom{\bar{R}}{0} \tag{4.5}$$

ALGORITMO

Passo 1 : Faça $k = p-1$

Passo 2 : Se $k = t-1$ então $\hat{R}$ é a matriz $\binom{\bar{R}}{0}$

Passo 3 : Faça $k = k+1$

Obter $P_{k+1}^{k} \hat{R}$

Obter $P_{k+1}^{k} \hat{r}_k$ (onde $\hat{r}_k$ é a k-ésima coluna de $\hat{R}$)

$a = ({r_{kk}}^2 + r_{k+1,k}^2)^{1/2}$ (onde r_{kk} é o (k,k) é-simo elemento de $\hat{R}$)

$c = r_{kk}/a$

$s = r_{k+1,k}/a$

$r_{kk} = a$

$r_{k+1,k} = 0$

$y = s/(1+c)$

Obter $P_{k+1}^{k} \hat{r}_i$ para $i = k+1, \ldots t-1$

$a = cr_{ki} + sr_{k+1,i}$

$r_{k+1,i} = y(r_{ki} + a) - r_{k+1,i}$

$r_{ki} = a$

Vá ao passo 2.

4.4. Triangularização de uma matriz triangular superior com um vetor A_p anexado

4.4.1. A_p vetor linha na última posição

Considere $\hat{R}$ uma matriz ((t+1)xt) em que uma linha é anexada numa matriz triangular superior (txt), conforme abaixo:

$$\hat{R} = \begin{array}{cccccc} x & x & x & x & x & x \\ & x & x & x & x & x \\ & & x & x & x & x \\ & & & x & x & x \\ & & & & x & x \\ & & & & & x \\ x & x & x & x & x & x \end{array} \tag{4.6}$$

Aplicando transformações de Givens na sequência indicada em (4.7) à matriz $\hat{R}$, obtemos uma matriz triangular superior $\bar{R}(t,t)$:

$$P_{t+1}^{t} \cdots P_{t+1}^{2} \, P_{t+1}^{1} \, R = \binom{\bar{R}}{0} \tag{4.7}$$

onde 0 é um vetor linha.

ALGORITMO:

Passo 1 : Faça $k = 0$

Passo 2 : Se $k = t$ então $\hat{R}$ é a matriz $\binom{\bar{R}}{0}$

Passo 3 : Faça $k = k+1$

Obter $P^k_{t+1}\hat{R}$

Obter $P^k_{t+1}\hat{r}_k$:

$a = (r^2_{kk} + r^2_{t+1,k})^{1/2}$

$c = r_{kk}/a$

$s = r_{t+1,k}/a$

$r_{kk} = a$

$r_{t+1,k} = 0$

$y = s/(1+c)$

Obter $P^k_{t+1}\hat{r}_i$ para $i = k+1, \ldots t$:

$a = cr_{ki} + sr_{t+1,i}$

$r_{t+1,i} = y(r_{ki}+a) - r_{t+1,i}$

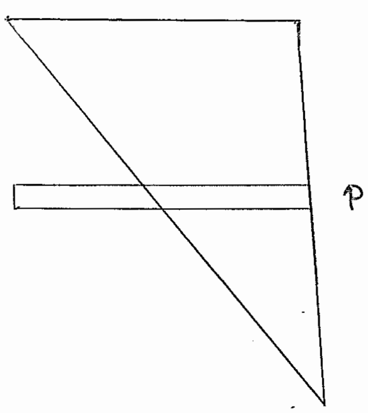
$r_{ki} = a$

Vá ao passo 2.

4.4.2. A_p vetor linha em posição intermediária

O caso de A_p ser um vetor linha anexado intermediário, é a situação mais frequente nos problemas de otimização. Geralmente acontece quando a matriz triangular é modificada tomando a forma $\hat{R} = R + e_pA_p$ como representado abaixo:

$\hat{R}$ =



(4.8)

Neste caso R é uma matriz triangular superior e A_p é um vetor linha que modifica o vetor linha R_p da matriz R.

Para transformar $\hat{R}$ numa matriz triangular superior, propõe-se dois métodos:

a) Usando Matrizes de Givens diretamente e

b) Usando uma matriz de permutação a fim de transformar a matriz (4.8) numa matriz Hessenberg e após aplicar matrizes de Givens para atingir a forma de uma matriz triangular superior.

O primeiro método será explicado nesta seção e o outro será exposto na seção (4.4.3) ao considerar A_p um vetor coluna.

4.4.2.1. <u>Uso de transformações de Givens</u>

Faz-se uma sequência de rotações dos planos (1,p). ... (p-1,p) até que o "bico" desapareça. Na matriz (4.8) $R + e_pA_p$, temos o vetor linha $\bar{R}_p = R_p + A_p$; onde $\bar{r}_{pk} \neq 0$ para $k = 1,2,\ldots p-1$, deve ser zerado. Portanto o procedimento consiste em calcular a matriz $\bar{R} = P_p^{p-1} \ldots P_p^1 \bar{R}$.

<u>ALGORITMO</u>:

Passo 1 : Faça $k = 0$

Faça $\bar{R} = R + e_pA_p$.

Passo 2 : Se $k = p-1$ então $\bar{R}$ é triangular superior.

Passo 3 : Faça $k = k+1$

Obter $P_p^k \bar{R}$:

Obter $P_p^k R_k$:

$a = (\bar{r}_{kk}^{\,2} + \bar{r}_{pk}^{\,2})^{1/2}$

$c = \bar{r}_{kk}/a$

$s = \bar{r}_{pk}/a$

$\bar{r}_{kk} = a$

$\bar{r}_{pk} = 0$

Obter $P_p^k R_i$ para $i = k+1,\dots,n$:

$a = c\ \bar{r}_{ki} + s\ \bar{r}_{pi}$

$b = -s\ \bar{r}_{ki} + c\ \bar{r}_{pi}$

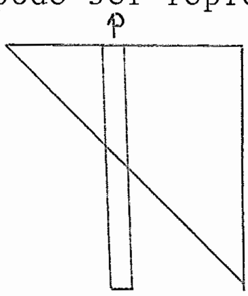
$\bar{r}_{ki} = a$

$\bar{r}_{pi} = b$

Vá ao Passo 2.

4.4.3. A_p vetor coluna em posição intermediária

No caso de ser A_p um vetor coluna anexado na matriz R , pode ser representado como $\hat{R} = R + A_p e_p^t$, onde:



$$\equiv R + A_p e_p^T \tag{4.9}$$

Na seção anterior (4.4.2), em que A_p era um vetor linha, mencionou-se duas maneiras para transformar (4.8) numa matriz triangular superior. Uma já foi exposta nessa seção e a outra será estudada agora.

O procedimento consiste em usar permutações dos vetores, neste caso vetores colunas para transformar (4.9) numa matriz de Hessenberg e daí usar transformações de Givens ou as matrizes de permutações para chegar-se a uma matriz triangular.

Aplicando-se a $\hat{R}$ uma matriz de permutação à direita conveniente, obtém-se uma forma de Hessenberg superior:

$$H = \hat{R}\, E_p = (R + A_p e_p^T)\, E_p \tag{4.10}$$

onde E_p é a matriz de permutação. A título de exemplo, se $\hat{R}$ possui a estrutura

$$\hat{R} = \begin{bmatrix} x & x & x & x & x & x \\ & x & x & x & x & x \\ & x & x & x & x & x \\ & x' & & x & x & x \\ & x' & & & x & x \\ & x' & & & & x \end{bmatrix} \tag{4.11}$$

basta considerar sucessivamente as permutações (2,3), (2,4) e (2,5) que "empurram" cada elemento não nulo x' para o lugar do elemento nulo mais à direita da respectiva linha. Neste caso temos (com $p = 2$):

$$E_p = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{e} \quad H = \begin{bmatrix} x & x & x & x & x & x \\ & x & x & x & x & x \\ & x & x & x & x & x \\ & & x & x & x & x \\ & & & x & x & x \\ & & & & x & x \end{bmatrix}$$

A transformação de H (4.10) numa matriz triangular superior pode ser feita pelo algoritmo apresentado na seção (4.3) deste capítulo.

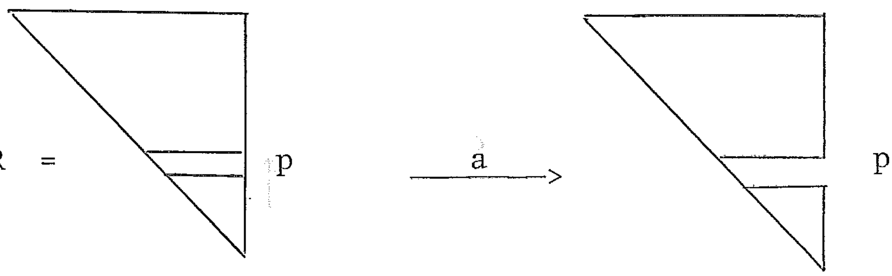
No caso, aplicando uma sucessão de rotações de planos, temos:

$$H = P_n^{n-1} \dots P_{p+2}^{p+1} \; P_{p+1}^{p} \; H \tag{4.12}$$

ESCUDERO (6) apresenta um algoritmo para transformar H (4.10) numa triangular superior baseando-se nos enfoques de TOMLIN (26) e FORREST e TOMLIN (12). Consiste em eliminar os elementos da linha p de H (4.10) nas colunas p até n e então realizar permutações de linhas.

O procedimento desse algoritmo chamado UTR ("Upper triangular restoration") é:

(1) Zerar a linha p da matriz R nas colunas p+1 até n :

De R = 

a) Isso é feito usando uma matriz T^{-1} ,

$$T^{-1} = I - e_p t \tag{4.13}$$

onde e_p é o p-ésimo vetor unitário, t é um vetor linha da forma:

$$t = (0,0,\dots 0,t_{p+1},\dots t_n) \tag{4.14}$$

e portanto a matriz T^{-1} tem a seguinte forma:

$$T^{-1} = \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & & 1, t_{p+1} & \cdots & -t_n \\ & & & & 1 \end{bmatrix} \qquad (4.15)$$

b) O vetor t é calculado por $t = r_p\, R^{-1}$, onde o vetor linha r_p , da forma $(0,\ldots 0, r_{p,p+1} \ldots r_{pn})$ é a p-ésima linha de R , a ser zerada.

c) Obtém-se o produto $T^{-1}\, R$:

$$T^{-1}\, R = (I - e_p t) R = R - e_p\, r_p\, R^{-1}\, R$$

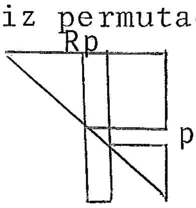
$$T^{-1}\, R = R - e_p r_p \qquad (4.16)$$

(2) Substituindo-se R na expressão (4.16) por $H = R + A_p e_p^T$ obtemos:

$$T^{-1}\, H = R + A_p e_p^T - e_p r_p \qquad (4.17)$$

Então considerando linhas j de H e de R temos $T^{-1}\, H_j \equiv T^{-1}\, R_j = R_j$ para $(j \neq p)$ e $j = 1,\ldots,n$. Portanto a matriz $T^{-1}\, H$ é a mesma matriz R cujos elementos $r_{p,p+1}, \ldots\, r_{pn}$ foram zerados, o vetor coluna R_p tendo sido substituido por $T^{-1}(R_p + A_p)$.

(3) $T^{-1}\, H$ é uma matriz permutada.

$T^{-1}\, H \equiv$  (4.18)

Podemos obter uma matriz $\overline{R}$ triangular superior:

$$\overline{R} = E_p^{-1}\, T^{-1}\, H\, E_p$$
$$\overline{R} = E_p^{-1}\, T^{-1}\, (R + A_p e_p^T)\, E_p \qquad (4.19)$$

onde E_p é uma matriz de permutação conveniente. O algoritmo detalhado pode-se encontrar em ESCUDERO (6).

CAPÍTULO V

FATORIZAÇÃO QR

5.1. Definição

A fatorização QR consiste em decompor uma matriz $A(nxt)$ como produto de duas matrizes $Q(nxn)$ e $R(nxt)$ onde a matriz Q é ortogonal e R é uma matriz triangular superior. Portanto:

$$A = QR \tag{5.1}$$

ou

$$Q^T A = R \tag{5.2}$$

Geralmente, a última expressão é mais usada já que Q não é obtida explicitamente. Esta matriz Q é formada por uma sequência de transformações ortogonais (ver Cap.III) cuja função é triangularizar A. Se A for de posto completo a fatorização QR é única.

De fato, a forma da matriz R varia, segundo as dimensões de A, o que será analisado na seção seguinte.

5.2. Fatorização QR segundo as dimensões de A

No Capítulo II foram apresentados vários casos de sistemas de equações segundo a estrutura da matriz A. Dentro deles os mais frequentes são:

5.2.1. Caso de sistema sobredeterminado $(n > t)$

Considerando uma matriz $A(nxt)$ e sendo $(n > t)$, a decomposição QR será da forma:

$$Q^T A = \binom{R}{0} \tag{5.3}$$

onde $Q(nxn)$ e R uma matriz triangular superior (txt) não necessariamente com diagonal unitária.

Ao se particionar Q segundo $Q = (Q_1, Q_2)$, onde Q_1 é uma matriz (nxt) e Q_2 é $(nx(n-t))$ obtemos:

$$\begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix} A = \binom{R}{0} \tag{5.4}$$

isto é:

$$Q_1^T A = R \qquad\qquad Q_2^T A = 0 \tag{5.5}$$

e

$$A = Q_1 R \tag{5.6}$$

Além disso, se se particiona A em (A_1, A_2), tal que A_1 tenha k colunas e R seja particionado segundo

$$R = \begin{bmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{bmatrix} \tag{5.7}$$

sendo R_{11} uma matriz (kxk) obtemos:

$$\begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix} (A_1 \ A_2) = \begin{bmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{bmatrix}$$

de onde:

$$Q_1^T A_1 = R_{11} \quad \text{e} \quad Q_2^T A_1 = 0 \tag{5.8}$$

Portanto Q e R_{11} compreende a fatorização QR de A_1. Isto é chamado fatorização QR truncada e é obtida sem custo adicional uma vez que a fatorização QR geral tenha sido feita.

5.2.1.1. Posto (A) = k < t

Neste caso a decomposição QR de A não é unica. DONGARRA et alii (4) propõem a utilização de uma matriz de permutação E tal que AE é particionada da forma:

$$AE = (A_1, A_2) \tag{5.9}$$

onde A_1 tem k colunas linearmente independentes. O objetivo evidente é separar uma matriz A_1(nxk) de posto completo.

A parte triangular da fatorização QR toma a forma:

$$Q_1^T AE = R = \begin{bmatrix} R_{11} & R_{12} \\ 0 & 0 \end{bmatrix} \tag{5.10}$$

onde R_{11} e R_{12} são únicos e A_2 é calculada usando (5.9) isto é:

$$Q^T A E = \begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix} (A_1 \; A_2) = \begin{bmatrix} R_{11} & R_{12} \\ 0 & 0 \end{bmatrix}$$

e então:

$$Q_1^T A_1 = R_{11} \tag{5.11}$$

$$Q_1^T A_2 = R_{12} \tag{5.12}$$

Usando (5.11) e (5.12) e sendo Q_1 ortogonal temos:

$$A_2 = A_1 R_{11}^{-1} R_{12} \tag{5.13}$$

o qual expressa as colunas restantes de AE como combinações lineares das colunas de A_1.

No trabalho de DONGARRA et alii (4) apresenta a subrotina SQRDC que possui uma opção para permutar as colunas de A tal que as colunas iniciais de AE sejam independentes. Este pivoteamento é feito paralelamente à realização da decomposição e a estratégia para selecionar as colunas é efetuada na matriz R, quem possui uma diagonal dominante i.e:

$$r_{kk}^{2} \geq \sum_{i=k}^{j} r_{ij} \quad (j=k,k+1,\ldots p) \tag{5.14}$$

Portanto, se r_{kk} é zero, as linhas k até p são também zero.

5.2.2. Caso de Sistema Indeterminado $(n < t)$

Neste caso, a fatorização QR toma a seguinte forma:

$$Q^T A = (R\ S) \tag{5.15}$$

onde R é uma matriz triangular superior (txt) . Se as primeiras t colunas de A são linearmente independentes, a decomposição é única. Ver equações (5.36), (5.41) e (5.42).

Novamente, o pivoteamento permite isolar um conjunto de colunas linearmente independentes. Então existe uma matriz de permutação tal que:

$$Q^T A E = (R\ S) \tag{5.16}$$

O pivoteamento obriga à matriz R a ser não singular se a matriz A tem um conjunto de colunas linearmente independentes. Neste caso as primeiras t colunas de AE formam também uma matriz não singular.

5.3. Fatorização QR para resolver o Problema dos Mínimos Quadrados

Como foi exposto na seção (2.2) do Capítulo II, o problema de Mínimos Quadrados consiste em minimizar o vetor residual r , onde:

$$||r||^2 = ||b-Ax||^2 \tag{5.17}$$

A condição necessária e suficiente para que X_L , seja solução de Mínimos Quadrados é:

$$A^T (b - AX_L) = 0 \tag{5.18}$$

O vetor x que é solução do sistema (5.18) é:

$$x = A^{+}b \tag{5.19}$$

onde: $A^{+} = (A^{T}A)^{-1} A^{T}$, se A tem posto completo (5.20)

A expressão (5.20) define a pseudo-inversa da matriz A, que não é conveniente calcular-se por ser inexata se A for mal condicionada.

Uma maneira de evitar o cálculo da Pseudo-inversa é fatorizar a matriz A:

$$Q^{T} A = \binom{R}{O} \tag{5.21}$$

onde Q^{T} é uma matriz ortogonal (nxn) e R é uma matriz triangular superior (txt), não necessariamente com diagonal unitária.

Como se pode apreciar, a decomposição (5.21) para a matriz A é a mesma que para o caso de sistemas sobredeterminado. Será também visto o caso de sistemas indeterminados.

Para encontrar uma expressão para x_L, solução do Problema dos Mínimos Quadrados considerando a fatorização QR da matriz A, particionaremos a matriz Q em:

$$Q = (Q_1 \; Q_2) \tag{5.22}$$

onde Q_1 é uma matriz (nxt) e Q_2 é uma (nx(n-t)), tais que $A = Q_1 R$.

Desde que A é uma matriz de posto completo i.e. posto(A) = min(n,t), obtemos:

a) R é uma matriz não singular.

b) $I = Q\ Q^T$, i.e.: (5.23)

$$(Q_1\ Q_2)\begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix} = Q_1\ Q_1^T + Q_2\ Q_2^T \tag{5.24}$$

$I = Q^T\ Q$, i.e.: (5.25)

$$\begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix} (Q_1\ Q_2) = \begin{bmatrix} Q_1^T\ Q_1 & Q_1^T\ Q_2 \\ Q_2^T\ Q_1 & Q_2^T\ Q_2 \end{bmatrix} \tag{5.26}$$

Portanto:

$$Q_1^T\ Q_1 = Q_2^T\ Q_2 = I \tag{5.27}$$

$$Q_1^T\ Q_2 = Q_2^T\ Q_1 = 0 \tag{5.28}$$

c) $Q^T\ A = \binom{R}{0}$ (5.29)

i.e.

$$\begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix} A = \binom{R}{0}$$

$$\begin{bmatrix} Q_1^T\ A \\ Q_2^T\ A \end{bmatrix} = \binom{R}{0}$$

Portanto:

$$Q_1^T\ A = R \tag{5.30}$$

$$Q_2^T\ A = 0 \tag{5.31}$$

d) A solução dos Mínimos Quadrados x_L é encontrada tal que minimize o residual $||r||^2 = ||b-Ax||^2$

$$\text{Min } ||r||^2 = ||b-Ax||^2 \tag{5.32}$$

Transformando o segundo membro da equação (5.32):

$$= ||Q^T b - Q^T Ax||^2 = \left\| \begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix} b - \binom{R}{0} x \right\|^2$$

$$= \left\| \begin{matrix} Q_1^T b - Rx \\ Q_2^T b \end{matrix} \right\|^2$$

Portanto, o Min $||r||^2 = \left\| \begin{matrix} Q_1^T b - Rx \\ Q_2^T b \end{matrix} \right\|$ será atingido

quando existe x_L tal que:

$$Q_1^T b = Rx_L$$

$$x_L = R^{-1} Q_1^T b \tag{5.33}$$

e Min $||r||^2 = ||Q_2^T b||$.

Uma outra maneira de obter o vetor residual é usando: (5.30) e (5.33),

$$r = b - A x_L$$

$$r = b - Q_1 R R^{-1} Q_1^T b$$

$$r = (I - Q_1 Q_1^T) b \tag{5.34}$$

Mais ainda, usando (5.24) i.e. $Q Q^T = I$ temos uma expressão para r, isto é:

$$r = Q_2 Q_2^T b \tag{5.35}$$

Porém, a equação (5.34) é mais vantajosa que (5.35) já que evita o cálculo de Q_2.

Considerando agora um sistema indeterminado, sabemos que a fatorização QR para a matriz A é da forma:

$$Q^T A = (R \; S) \tag{5.36}$$

onde Q é uma matriz ortogonal $(n x n)$, R é uma matriz triangular superior $(n x n)$ e S uma matriz $(n x (t-n))$.

Na seção (5.2.2) viu-se que a decomposição (5.36) não é a única a menos que as primeiras n colunas da matriz A sejam linearmente independentes. Para isso a matriz A será particionada em:

$$A = (A_1 \; A_2) \tag{5.37}$$

onde A_1 é uma matriz quadrada cujas n colunas são linearmente independentes. Neste caso A_1 e R são matrizes não singulares.

Usando a equação (5.36) nas equações normais:

$$A^T A x = A^T b$$

obtemos:

$$\begin{bmatrix} R^T \\ S^T \end{bmatrix} Q^T Q \, (R \; S) x = \begin{bmatrix} R^T \\ S^T \end{bmatrix} Q^T b \tag{5.38}$$

de onde:

$$R^T (R \; S) x = R^T Q^T b \tag{5.39}$$

$$(R \; S) x = Q^T b \tag{5.40}$$

Por outro lado, sendo $A = (A_1\ A_2)$ temos de (5.36) o seguinte:

$$Q^T A = (R\ S)$$

$$Q^T(A_1\ A_2) = (R\ S)$$

$$Q^T A_1 = R \tag{5.41}$$

$$Q^T A_2 = S \tag{5.42}$$

Da equação (5.41):

$$Q^T = R A_1^{-1} \tag{5.43}$$

Substituindo (5.43) em (5.42):

$$S = R A_1^{-1} A_2 \tag{5.44}$$

Em resumo, voltando à equação (5.40) e usando uma decomposição conveniente $x = (x_1^T, x_2^T)^T$ obtemos:

$$(R\ S) \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = Q^T b \tag{5.45}$$

i.e.

$$R x_1 + S x_2 = Q^T b$$

$$R x_1 = Q^T b - S x_2$$

$$R x_1 = Q^T b - R A_1^{-1} A_2 x_2 \tag{5.46}$$

Se x_2 se anula (a solução de norma mínima) os primeiros n elementos da solução dos Mínimos Quadrados x_L ,i.e.

o vetor x_1 , é encontrado resolvendo-se

$$R\ x_1 = Q^T\ b \tag{5.47}$$

Como (5.32), a equação (5.47) é mais exata que o cálculo da pseudo-inversa, obtendo-se:

$$\begin{aligned} x_{1n} &= Q_n^T\ b\ /\ R_{nn} \\ x_{1i} &= (Q_i^T\ b - \sum_{r=i+1}^{n} R_{ir}\ x_{1r})/R_{ii} \end{aligned} \tag{5.48}$$

para i = n-1 a 1

onde Q_i é o i-ésimo vetor coluna da matriz Q .

CAPÍTULO VI

FATORIZAÇÃO QR PARA SISTEMAS SOBRE DETERMINADOS

Apresentaremos dois procedimentos para obter Q na fatorização QR, ambos baseados no Método Modificado de Gram-Schmidt (MGS).

6.1. Cálculo das Matrizes Q e R

A matriz Q(nxt) é calculada em t iterações recursivas. Chamamos $Q^{(o)} = A$, e, ao final da iteração k, a matriz $Q^{(k)}$ tem a seguinte expressão:

$$Q^{(k)} = (q_1^{(k)}, \dots\, q_k^{(k)}, \dots\, q_t^{(k)}) \tag{6.1}$$

onde $q_i^{(k)}$ é o i-ésimo vetor coluna da matriz $Q^{(k)}$. Os vetores $q_i^{(k)}$, $i = 1,\dots,k$, são os vetores definitivos $q_i^{(k)} \equiv q_i^{(t)}$, $i = 1,\dots\, k$ da matriz $Q \equiv Q^{(t)}$.

Portanto, na iteração k, os vetores $q_i^{(k)}$ $i=1,\dots\, k$ são ortonormais:

$$q_i^{(k)^T} q_j^{(k)} = \begin{cases} 1 & i=j \\ 0 & i \neq j \end{cases} \quad \text{para } i,j=1,2,\dots,k \tag{6.2}$$

A matriz R é calculada de maneira similar com $R_{ij} = 0$ para $i > j$ e $i,j = 1,2,\dots\, t$. O vetor linha R_k

da matriz R é obtida na mesma iteração k em que o vetor q_k é obtido.

6.2. Procedimento para obter a matriz Q

Apresentamos dois algoritmos para obter a matriz Q. A diferença entre eles está em a matriz Q ser normalizada ou não. Esse fato de normalizar ou não a matriz influencia na estabilidade do procedimento.

Considere por exemplo as matrizes Q, R, Q' e R' tal que:

$$A = Q' R' = QR \qquad (6.3)$$

onde Q é uma matriz não normalizada. Veremos que Q' e Q se relacionam através de:

$$Q' = Q D^{-1} \qquad (6.4)$$

onde D é uma matriz diagonal (txt) com seus elementos:

$$D(i,i) = d_i^{1/2} \qquad i = 1, \dots t \qquad (6.5)$$

$$d_i = ||q_i||_2^2 = \sum_{h=1}^{n} q_{hi} \qquad (6.6)$$

Então, $Q^T Q = D^2$

Com efeito, de (6.4),

$$Q = Q' D$$

$$Q^T = D\ Q'^T$$
$$Q^T\ Q = D\ Q'^T\ Q'\ D$$

Por ser Q' uma matriz ortonormal, temos $Q^T\ Q = D^2$.

Da mesma forma R é uma matriz triangular superior com diagonal unitária, e se tem

$$R' = D\ R \tag{6.7}$$

De tudo que foi visto, temos que usando as matrizes Q e R no lugar de Q' e R', as operações de raízes quadradas são evitadas causando aparentemente vantagens de ordem computacional.

6.2.1. Algoritmo Q Normalizado (N Q A)

Passo 1 : Obter as normas

$$N_i^{(k)} = \left[\sum_{h=1}^{n} q_{hi}^{(k)} \right]^{1/2} \tag{6.8}$$

para $i=k,k+1,\dots t$

Passo 2 : Seja $N_{k'}^{(k)}$ o máximo das normas no passo anterior. Se existem diferentes normas com o mesmo valor máximo, arbitrariamente escolhe-se a norma com o primeiro subíndice. Desde que se assume que A é de posto completo temos que $N_{k'}^{(k)} \neq 0$.

Passo 3 : Se $k' > k$, as k'-ésimas colunas da matriz $Q^{(k)}$ são trocadas. Também trocam-se os elementos $R_{ik'}$ e R_{ik} dos vetores linha R_i (para $i=1,2,\ldots k-1$).

Após a mudança continua-se chamando $q_i^{(k)}$ à i-ésima coluna da matriz $Q^{(k)}$ e R_i à i-ésima linha da matriz R.

Passo 4 : Obter o vetor q_k para a iteração $(k+1)$ o qual será o vetor definitivo q_k da matriz Q

$$q_k^{(k+1)} = q_k^{(k)} / N_k^{(k)} \tag{6.9}$$

Passo 5 : Obter os elementos R_{ki} $(i=k,k+1,\ldots t)$ do vetor linha R_k .

$$R_{kk} = q_k^{(k+1)^T} q_k^{(k)}$$

$$R_{kk} = q_k^{(k)^T} q_k^{(k)} / N_k^{(k)} = N_k^{(k)^2} / N_k^{(k)}$$

$$R_{kk} = N_k^{(k)} \tag{6.10}$$

$$R_{ki} = q_k^{(k+1)^t} q_i^{(k)}$$

$$R_{ki} = q_k^{(k)^T} q_i^{(k)} / N_k^{(k)} \tag{6.11}$$

para $i=k+1,\ldots t$

<u>Passo 6</u> : Calcular vetores q para a iteração (k+1)

$$q_i^{(k+1)} = q_i^{(k)} - R_{ki}\ q_k^{(k+1)} \tag{6.12}$$

para $i=k+1,\ldots t$

A ortogonalidade da expressão (6.2) implica que os vetores colunas $q_i^{(k)}$ $(i=1,2,\ldots k)$ são teoricamente normalizados no final da iteração k(ie. $\sum_{h=1}^{n} q_{hi}^{(k)^2}$ deve ser igual a 1). Mas o cálculo de $q_k^{(k+1)}$ não é exata ,dados os cálculos das normas que não são exatas, devido à representação limitada de números de ponto flutuante no computador. Mais ainda, essa instabilidade é incrementada no cálculo de (6.12).

Por tudo isto, a fim de evitar essa instabilidade, propõe-se uma ligeira modificação no algoritmo anterior. Consiste em não realizar o passo 4 (i.e. não calcular o vetor q_k para a iteração (k+1)) e armazenar no escalar d_k a norma $N_k^{(k)^2}$ que foi calculada nos passos 2 e 3. Ao final do algoritmo modificado, o vetor coluna d(tx1) armazena as normas dos vetores $q_1, q_2, \ldots q_t$ que já não estão normalizados.

Portanto, a expressão de ortogonalidade da matriz será:

$$(q_i^T\ q_j)/(N_i\ N_j) = \begin{cases} 1 & i=j \\ 0 & i\neq j \end{cases} \tag{6.13}$$

para $i,j=1,2,\ldots t$ isto é:

$$Q^T\ Q\ \delta^T\ \delta = I \tag{6.14}$$

onde o i-ésimo elemento do vetor coluna $(t\mathrm{x}1)\delta$ é $1/d_i^{1/2}$.

6.2.2. Algoritmo Q Não Normalizado (NNQA)

Na iteração k deste algoritmo temos:

Passo 1 : Obter os elementos $d_i (i=k,k+1,\ldots t)$ do vetor d

$$d_i = {N_i^{(k)}}^2 = \sum_{h=1}^{n} {q_{hi}^{(k)}}^2 \qquad (6.15)$$

Passo 2 : Análogo ao passo 2 do algoritmo NQA

Passo 3 : Análogo ao passo 3 do algoritmo NQA

Passo 4 : Fazer:

$$q_k^{(k+1)} = q_k^{(k)}$$

Passo 5 : Obter os elementos R_{ki} $(i=k,k+1,\ldots t)$ do vetor linha R_k , mas agora normalizando-a com $N_k^{(k)}$, isto é:

$$R_{kk} = (({q_k^{(k)}}^t q_k^{(k)})/N_k^{(k)})/N_k^{(k)}$$

$$R_{kk} = {N_k^{(k)}}^2/{N_k^{(k)}}^2 \quad (\Rightarrow \quad R_{kk} = 1) \qquad (6.16)$$

$$R_{ki} = (({q_k^{(k)}}^T q_i^{(k)})/N_k^{(k)})/N_k^{(k)}$$

(Mas sabemos que $d_k = {N_k^{(k)}}^2$ e $q^{(k+1)} = q^{(k)}$)

então:

$$R_{ki} = ({q_k^{(k+1)}}^T q_i^{(k)})/d_k \qquad (6.17)$$

Passo 6 : Obter vetores q para a iteração (k+1) . Da mesma forma que no algoritmo NQA , será calculado o vetor $q_i^{(k+1)}$ mas agora considerando que este vetor não está normalizado. Sendo que o vetor R_{ki} será normalizado. Portanto:

$$q_i^{(k+1)} = q_i^{(k)} - R_{ki}\, N_k^{(k)} (q_k^{(k+1)}/N_k^{(k)})$$

$$q_i^{(k+1)} = q_i^{(k)} - R_{ki}\, q_k^{(k+1)} \tag{6.18}$$

para i=k+1,... t

Pode-se observar que no algoritmo NNQA não está sendo usado o valor de $N_k^{(k)}$ senão o valor d_k , portanto, tem-se poucos erros de arredondamento.

Embora no algoritmo NNQA calcula-se as matrizes Q e R onde as colunas de Q não são normalizadas e sim as linhas da matriz R , a fatorização QR é a mesma. Isto é, chamando Q e R as matrizes calculadas pelo algoritmo NNQA e Q', R' as matrizes calculadas pelo algoritmo NQA, temos:

$$Q'\, R' = Q\, R \tag{6.19}$$

O anterior pode ser provado considerando o elemento A_{ij} da matriz A o qual pode ser escrito como:

$$A_{ij} = \sum_{h=1}^{j} Q'_{ih}\, R'_{hj} = \sum_{h=1}^{j} (Q_{ih}/N_i)(R_{hj}\, N_j)$$

$$A_{ij} = \sum_{h=1}^{j} Q_i\, R_j$$

onde $R_{jj} = 1$.

No trabalho de ESCUDERO (5) afirma-se que ao obter a matriz Q não normalizada e R uma matriz normalizada com a diagonal unitária i.e. pelo algoritmo NNQA, o procedimento é mais estável que no caso em que Q seja normalizada e R não normalizado (algoritmo NQA).

6.3. Cálculo do Vetor Residual r = b-Ax

Uma vez obtida a fatorização QR da matriz A,com as matrizes Q e R calculadas pelo algoritmo NNQA, é possível obter o valor de x . Isto será feito com a equação (5.32) a qual baseia-se nas propriedades de ortogonalidade da matriz Q isto é:

$$R\, x_L = Q^T\, b$$

Acontece que no cálculo da matriz Q , existem erros de arredondamento que poderiam fazer perder as propriedades de ortogonalidade e mais ainda o valor de x seria instável. Portanto, uma maneira de evitar essa possibilidade de erro, é substituir em qualquer operação intermediária a expressão

$$q_i^T\, q_j = 0 \qquad \text{para} \quad i \neq j$$

que na realidade é o valor teórico.

A idéia do procedimento é obter primeiro o vetor residual r e após o vetor de incógnitas x, sendo que o vetor residual $r(nx1)$ é obtido em t-iterações de um modo recursivo. Para isto considera-se $r^{(o)} = b$ e no final da iteração t, o residual $r^{(t)}$ toma o valor do vetor residual r.

Considerando a equação (5.34), a ortogonalidade da matriz Q' i.e. $Q' Q'^T = I$ e a notação $Q_1 = Q$ obtemos:

$$r = (I - Q_1 Q_1^T)b$$

$$r = b - Q' Q'^T b$$

$$r = b - \sum_{i=1}^{t} (q_i'^T b)q_i'$$

$$r = b - \sum_{i=1}^{t-1} ((q_i^T b)/d_i)q_i - ((q_t^T b)/d_t)q_t \qquad (6.20)$$

Definindo:

$$r^{(t-1)} = b - \sum_{i=1}^{t-1} ((q_i^T b)/d_i)q_i \qquad (6.21)$$

e substituindo-se (6.21) em (6.20) obtemos:

$$r = r^{(t-1)} - ((q_t^T b)/d_t)q_t \qquad (6.22)$$

Analogamente ao anterior, a expressão (6.22) pode ser substituida por:

$$r^{(k)} = r^{(k-1)} - ((q_k^T b)/d_k)q_k \qquad (6.23)$$

onde:

$$r^{(k-1)} = b - \sum_{i=1}^{k-1} ((q_i^T\ b)/d_i)q_i \qquad (6.24)$$

para $k=1,2,\ldots\ t$

tal que $r^{(k)}$ para $k=t$ é o vetor resitual r .

Na realidade, a estabilidade do procedimento anterior depende da exatidão do cálculo da matriz Q pelo algoritmo NNQA . Como já foi exposto, os erros de arredondamento durante o processo, podem fazer perder a ortogonalidade da matriz Q i.e. pode-se dar que $q_k^t\ q_i \neq 0$ originando que o valor de $r^{(k)}$ seja inexato.

Portanto, a expressão $q_k^T\ q_i$ será substituida por zero, obtendo uma nova expressão para (6.24). Mas, convém trabalhar antes com a dita expressão, multiplicando-a por (q_k^T/N_k) a fim de amostrar o anterior e fazer a substituição respectiva: $(q_k^T/N_k)\,r^{(k-1)} = (q_k^T/N_k)b - (q_k^T/N_k) \sum_{i=1}^{k-1} ((q_i^Tb)/N_i)(q_i/N_i)$ desde que $d_i = N_i^2$.

$$(q_k^T\ r^{(k-1)})/N_k = (q_k b/N_k) - \sum_{i=1}^{k-1}((q_i^T\ q_i)/N_i)((q_k^T\ q_i)/N_i) \qquad (6.25)$$

Agora substituindo-se $q_k^T\ q_i = 0$ em (6.25),temos:

$$(q_k^T\ r^{(k-1)})/N_k = (q_k^T\ b)/N_k$$

$$q_k^T\ r^{(k-1)} = q_k^T\ b \qquad (6.26)$$

Ao final, obtemos uma expressão para $r^{(k)}$:

$$r^{(k)} = r^{(k-1)} - ((q_k^T\ b)/d_k)q_k$$

$$r^{(k)} = r^{(k-1)} - ((q_k^T\ r^{(k-1)}/d_k)q_k \qquad (6.27)$$

para $k=1,2,\ldots\ t$

e $r^{(0)} = b$

$r = r^{(t)}$

6.4. Cálculo do Vetor de Incógnitas x

Para calcular o valor de x, que minimiza $||r||^2$, pode-se usar a expressão (5.33) i.e. $R'\ x = Q'^T\ b$; mas considerando o elemento x_k temos:

$$R'_{kk}\ x_k + \sum_{i=k+1}^{t} R'_{ki}\ x_i = q_k'^T\ b \qquad (6.28)$$

para $k=1,2,\ldots\ t-1$

$$R'_{tt}\ x_t = q_t'^T\ b\ .$$

Agora, o objetivo é obter as matrizes Q e R como no algoritmo NNQA a fim de obter uma expressão para x.

$$N_k\ x_k = (q_k^T\ b)/N_k - \sum_{i=k+1}^{t} N_k\ R_{ki}\ x_i \qquad (6.29)$$

para $i=1,2,\ldots\ t-1$

e

$$N_t\ x_t = (q_t^T\ b)/N_t$$

Mas, desde que $q_k^T\ r^{(k-1)} = q_k^T\ b$ ver (6.26) temos:

$$x_k = (q_k^t\ r^{(k-1)})/d_k - \sum_{i=k+1}^{t} R_{ki}\ x_i \tag{6.30}$$

para $i=1,2,\ldots,\ t-1$

onde $d_k = N_k^2$, e:

$$x_t = (q_t\ r^{(t-1)})/d_t \tag{6.31}$$

No cálculo de $r^{(k)}$ e de $x^{(k)}$, usa-se as mesmas operações intermediárias e pode-se evitar novos cálculos de raízes quadradas que produzem muitos erros de arredondamento.

CAPÍTULO VII

ATUALIZAÇÕES DA FATORIZAÇÃO QR

No capítulo IV ao serem apresentadas a triangularização de matrizes por transformações de Givens, mencionou-se a relação com os tópicos a serem desenvolvidos neste capítulo.

O objetivo é apresentar diferentes critérios para as atualizações da fatorização QR os quais fornecem as bases teóricas para modificar a matriz de Restrições Ativas nos problemas de Programação Não Linear que são resolvidos segundo a metodologia da estratégia de Conjuntos Ativos.

Os tópicos desenvolvidos neste capítulo, encontram-se atualmente disseminados em diversos artigos e livros. Tentanto-se aqui fornecer uma visão geral deles.

A atualização da fatorização QR pode ser feita seguindo o critério de Modificação de Posto Um ou os Critérios de Adição ou Apagamento de linha ou coluna. Posteriormente os aprofundaremos com ênfase no último.

Ao modificar os fatores deve-se considerar os seguintes problemas:

a) A modificação deve ser feita em poucas operações, o quanto seja possível, especialmente para grandes sistemas onde se precisa de atualizações contínuas.

b) O procedimento numérico deve ser estável.

c) Se a matriz original é esparsa, convém manter a esparsidade tanto quanto seja possível.

No capítulo anterior viu-se um algoritmo de fatorização QR e apresentaram-se dois procedimentos para obter a matriz Q . Como resultado, usando o procedimento NQA obtemos as matrizes Q' e R' em que as colunas de Q' são normalizadas, e, usando o procedimento NNQA, obtemos as matrizes Q e R , onde agora as linhas de R são normalizadas.

Temos,portanto:

$$A = Q'R' = QR \tag{7.1}$$

Mais ainda, lembrando (6.19) e (5.30) sabemos que:

$$A = Q_1' \, R' = Q_1 \, R$$

em que Q_1 é a matriz não normalizada e Q_1' é a normalizada, i.e:

$$Q_1' = Q_1 \, D^{-1} \tag{7.2}$$

onde D é uma matriz diagonal tal que:

$$D(i,i) = d_i^{1/2} \tag{7.3}$$

e $\quad d_i = ||q_i||_2^2 = \sum_{h=1}^{n} q_{hi} \qquad i = 1,2,\ldots,t$.

Por outro lado, considerando a matriz D^2 , $(D^2(i,i) = d_i)$, temos em Q_1 uma matriz ortogonal com:

$$Q_1^T \, Q_1 = D^2 \tag{7.4}$$

Da mesma forma, R é uma matriz triangular superior com diagonal unitária, tal que:

$$R' = DR \tag{7.5}$$

Uma vez obtidos os fatores Q_1 e R , admita-se a matriz A modificada, seja adicionando seja apagando uma linha ou uma coluna. Neste caso, forma-se uma nova matriz $\overline{A}$ e se procura novos fatores $\overline{Q}$ e $\overline{R}$. Estes devem ser calculados poupando esforço computacional i.e., através de atualizações nas matrizes Q e R onde foram selecionadas as linhas ou colunas a serem adicionadas ou apagadas.

A idéia geral consiste em apagar ou adicionar uma linha ou uma coluna às matrizes Q e R , de modo que $\overline{A} = \overline{Q}_1 R$; e aplicando transformações de Givens, modifica-se $\overline{Q}_1$ e $\overline{R}$ afim que sejam a fatorização $\overline{QR}$ da matriz $\overline{A}$.

Nos últimos anos, muitos pesquisadores, têm-se dedicado aos algoritmos de atualização, por exemplo GILL et alii (14) atualizam Q' e R' , DANIEL et alii (3) atualizam Q_1' e R' e ESCUDERO (9) apresenta algoritmos para atualizar Q_1 e R.

7.1. Modificação de uma matriz ortogonal

Antes de aprofundar nas atualizações da fatorização QR, daremos uma explicação da modificação de uma matriz ortogonal seguindo os linhamento de DANIEL et alii (3). Este processo será usado na seção (7.3) quando se adiciona um vetor coluna à matriz A.

Seja a matriz:

$$Q = (q_1 \; q_2 \; \dots \; q_n) \in R^{m\times n} \quad (m \geq n)$$

com colunas ortonormais tal que $Q^TQ = I_n$ e seja $v \varepsilon \mathbb{R}^m$. A idéia do processo é procurar vetores $q \varepsilon \mathbb{R}^m$, $r \varepsilon \mathbb{R}^n$ e um escalar ρ, tais que:

$$(Q,v) = (Q,q) \begin{bmatrix} I & r \\ 0 & \rho \end{bmatrix} \tag{7.6}$$

$$Q^Tq = 0 \tag{7.7}$$

a última coluna é portanto:

$$v = Qr + q\rho \tag{7.8}$$

Multiplicando (7.8) por Q^T, considerando que a matriz Q é ortonormal, e (7.7), obtemos:

$$Q^Tv = Q^TQr + Q^T q \rho$$

$$Q^T v = r \tag{7.9}$$

Fazendo $v' = q \rho$ e usando (7.8): (7.10)

$$v' = v - Qr \tag{7.11}$$

$$v' = v - Q Q^T v$$

$$v' = (I - Q Q^T)v \tag{7.12}$$

Se $m > n$, deve-se ter $||q|| = 1$, portanto:

$$\rho = ||(I - Q Q^T)v|| \tag{7.13}$$

i.e. $\rho = ||v'||$ tal que:

$$q = \frac{(I - Q Q^T)v}{\rho} \quad \text{ou} \quad q = \frac{v'}{\rho} \quad \text{se} \quad \rho \neq 0 \tag{7.14}$$

Quando $\rho = 0$, temos diretamente $v = Qr$ (da expressão (7.8)) e $(Q,v) = Q(I,r)$ tem posto n .

Se o processo falha, isto é, $\rho \cong 0$ (pode ser originado por erros de arredondamento no cálculo de Q , tal que a condição de ortogonalidade $Q^T q = 0$ não é satisfeita), teoricamente qualquer vetor unitário ortogonal ao posto de Q poderia ser substituido por q .

Se o processo de modificação é realizada na presença de erros de arredondamento, é improvável que ρ desapareça exatamente ficando apenas com um valor bastante pequeno. Este processo visa forçar que $Q^T v'$ seja pequeno relativamente a $||v||$.

Mas mesmo que $||Q^T v'|| = \varepsilon ||v||$ com um ε pequeno e se ρ é pequeno demais, então o vetor normalizado $q = v'/\rho$ satisfaria só $||Q^T q|| = \varepsilon ||v||/\rho$ e o erro relativamente a $||v||$ poderia ser enorme. A partir daí, ter-se-ia uma perda de ortogonalidade no vetor q calculado.

Para resolver o problema anterior, usa-se Reortogonalização dado que se $||v'||/||v||$ é pequeno, então a cancelação numérica terá provavelmente acontecidoao calcular v' , já que tanto v' quanto q são relativamente incertos em suas normas. Portanto, a idéia é corrigir v' por reortogonalização, isto é, aplicando-se o processo novamente, substuindo-se v por v'; assim obtém-se:

$$s = Q'v'$$

$$v'' = v' - Qs = v - Q(r+s) \qquad (7.15)$$

Comparando a expressão (7.11) e (7.15) vemos que

v' foi substituido por v'' e r por $(r+s)$.

Se $||v''||/||v||$ já não é tão pequeno, então v'' deve ser "escalado" a um vetor q satisfatório. De outro modo, o processo é repetido.

7.2. Modificação de Posto Um

Um problema é descrito como a atualização dos fatores de A segundo uma modificação de posto Um, quando a matriz modificada $\bar{A}$ toma a seguinte forma:

$$\bar{A} = A + \alpha\, y\, z^T \tag{7.16}$$

onde α é uma escalar e y , z são vetores. (A matriz $\alpha\, y\, z^T$ é de posto um).

A seguir apresentaremos uma atualização geral de posto um seguindo as idéias e simbologias de DANIEL et alii(3) (Observe que neste artigo se trabalha com a matriz Q normalizada).

Seja $A = QR \in R^{mxn}$ $(m > n)$, $u \in R^n$, $v \in R^m$ e a matriz modificada $\bar{A}$ i.e:

$$\bar{A} \cong A + v\, u^T$$

$$\bar{A} \cong (Q,v) \begin{bmatrix} R \\ u^T \end{bmatrix}$$

1) Aplica-se o processo de modificação (seção(7.1)) na equação (7.6) temos:

$$(Q,v) = (Q,q) \begin{bmatrix} I & r \\ o & \rho \end{bmatrix} \quad , \; Q^T q = 0 \; , \; ||q|| = 1$$

Então:

$$\bar{A} = (Q,q) \begin{bmatrix} I & r \\ 0 & \rho \end{bmatrix} \begin{bmatrix} R \\ u^T \end{bmatrix}$$

$$\bar{A} = (Q,q) \begin{bmatrix} R + r\,u^T \\ \rho u^T \end{bmatrix}$$

$$\bar{A} = (Q,q) \left[\begin{bmatrix} R \\ 0^T \end{bmatrix} + \begin{bmatrix} r \\ \rho \end{bmatrix} u^T \right]$$

onde $\tilde{Q} = (Q,q)$ e $\tilde{R} = \left[\begin{bmatrix} R \\ 0^T \end{bmatrix} + \begin{bmatrix} r \\ \rho \end{bmatrix} u^T \right]$

$$\bar{A} \cong \tilde{Q}\,\tilde{R}\ ,\ \tilde{Q} \text{ com colunas ortonormais} \tag{7.17}$$

2) Escolhe-se matrizes de Givens a serem aplicadas às matrizes na expressão (7.17), que transformarão $\tilde{R}$ numa forma de Hessemberg.

Sejam as matrizes $G^1_{n+1}\ G^1_n \ldots G^1_2$ tais que:

$$G\binom{r}{\rho} = G^1_2\ G^1_3 \ldots G^1_n\ G^1_{n+1}\binom{r}{\rho} = \gamma e_1 \tag{7.18}$$

Na realidade as matrizes G^1_{i+1} $(i = n, n-1, \ldots 1)$ introduzem zeros sucessivamente num vetor desde o último elemento até o segundo dado que G^1_{i+1} elimina o elemento $i+1$ da coluna 1. Aplicando Givens a uma matriz $(n+1) \times n$, obtemos;

$$G\begin{bmatrix} R \\ 0^T \end{bmatrix} = G^1_2 \ldots G^1_n\ G^1_{n+1}\begin{bmatrix} R \\ 0^T \end{bmatrix} \cong R' \tag{7.19}$$

Aplicaremos agora as matrizes de Givens a $\tilde{R}$, da expressão (7.17):

$$G\,\tilde{R} = G\left[\begin{bmatrix} R \\ 0^T \end{bmatrix} + \begin{bmatrix} r \\ \rho \end{bmatrix} u^T \right]$$

$$G\,\tilde{R} = G \begin{bmatrix} R \\ 0^T \end{bmatrix} + G \begin{bmatrix} r \\ \rho \end{bmatrix} u^T$$

$$G\,\tilde{R} = R' + \gamma\, e_1\, u^T \cong \hat{R} \tag{7.20}$$

Como resultado dessa aplicação, obtemos uma matriz $\hat{R}$ que é uma forma de Hessenberg superior.

Deve-se também aplicar Givens a $\tilde{Q}$, sendo que:

$$\tilde{Q}\, G^T = \tilde{Q}\, G^1_{n+1} \cdots G^1_2 = \hat{Q} \tag{7.21}$$

onde a matriz $\hat{Q}$ tem colunas ortonormais e $\bar{A}$ é: $\bar{A} = \hat{Q}\hat{R}$ (7.22)

3) Uma nova aplicação de matrizes de Givens se faz necessário, a fim de eliminar os elementos subdiagonais de $\hat{R}$.

Sejam agora as matrizes $H^1_2\, H^2_3 \cdots H^n_{n+1}$ onde H^i_{i+1} elimina o elemento (i+1) da coluna i que ao serem aplicadas na matriz $\hat{R}$, a tranformarão, numa matriz triangular superior $\bar{R}$ isto é:

$$H\,\hat{R} = H^n_{n+1} \cdots H^1_2\, \hat{R} = \begin{bmatrix} \bar{R} \\ 0^T \end{bmatrix} \tag{7.23}$$

$$\hat{Q}\, H^T = \hat{Q}\, H^1_2 \cdots H^n_{n+1} \cong (\bar{Q},\ \bar{q}) \tag{7.24}$$

onde $(\bar{Q},\ \bar{q})$ tem colunas ortonormais. Obtém-se finalmente:

$$\bar{A} = (\bar{Q},\ \bar{q}) \begin{bmatrix} \bar{R} \\ 0^T \end{bmatrix}$$

$$\bar{A} = \bar{Q}\,\bar{R} \tag{7.25}$$

7.3. Adição de um vetor coluna à matriz A

Dada a matriz $A(nxt)$ de posto completo $(n \geq t)$ com uma decomposição de QR, onde Q é uma matriz ortogonal (nxt) e R é uma matriz triangular superior (txt). Adicionamos uma coluna a_{t+1} à matriz A, teremos então uma matriz modificada $\bar{A}$ com:

$$\bar{A} = (A,\ a_{t+1}) = (Q\ R, a_{t+1})$$

$$\bar{A} = (Q,\ a_{t+1})\begin{pmatrix} R & 0 \\ & 1 \end{pmatrix}$$

$$\bar{A} = \tilde{Q}\ \bar{R} \tag{7.26}$$

Neste caso $\bar{R}$ é uma matriz triangular superior com diagonal unitária, mas $\tilde{Q}$ é uma matriz $(nx(t+1))$ que já não é completamente ortogonal dado que não se garante que:

$$Q^T\ a_{t+1} = 0 \tag{7.27}$$

Portanto, o processo consiste em transformar $\tilde{Q}$ numa matriz ortogonal $\bar{Q}$ tal que $\bar{R}$ mantenha suas características. Dado que a matriz Q é ortogonal, usaremos a atualização já exposta na seção (7.1) afim de encontrar um vetor $q(nx1)$ e um vetor $r(tx1)$ tais que:

$$(Q,\ a_{t+1}) = (Q\ q)\begin{pmatrix} I & r \\ & 1 \end{pmatrix} \quad e \quad Q^T\ q = 0 \tag{7.28}$$

Aplicando o anterior na expressão da matriz modificada $\bar{A}$, obtemos:

$$\bar{A} = (A,\ a_{t+1})$$

$$\bar{A} = (Q,\ a_{t+1}) \begin{pmatrix} R & 0 \\ & 1 \end{pmatrix}$$

$$\bar{A} = (Q\ q) \begin{pmatrix} I & r \\ & 1 \end{pmatrix} \begin{pmatrix} R & 0 \\ & 1 \end{pmatrix}$$

$$\bar{A} = (Q\ q) \begin{pmatrix} R & r \\ & 1 \end{pmatrix}$$

$$\bar{A} = \bar{Q}\ \bar{R} \tag{7.29}$$

Da expressão (7.28) segue que:

$$a_{t+1} = Q\ r + q \tag{7.30}$$

sendo que o vetor q é não normalizado. Denotando $\bar{Q} = (Q\ q)$, temos:

$$\bar{Q}^T\ \bar{Q} = (Q\ q)^T\ (Q\ q) = \bar{D}^2 \tag{7.31}$$

De acordo com a expressão (7.30) de a_{t+1} devemos obter os vetores r e q. Assumindo que $n > t$, pré-multiplica-se a expressão (7.30) por Q^T e considerando (7.28) e a condição de que Q é ortogonal obtemos:

$$Q^T\ a_{t+1} = Q^T(Qr + q)$$

$$Q^T\ a_{t+1} = Q^T\ Qr + Q^T\ q$$

$$Q^T\ a_{t+1} = D^2\ r$$

$$r = D^{2^{-1}}\ Q^T\ a_{t+1} \tag{7.32}$$

Da mesma forma, para o cálculo de q, considera-se (7.30) e (7.32), isto é:

$$q = a_{t+1} - Qr$$

$$q = a_{t+1} - Q(D^2)^{-1} Q^T a_{t+1}$$

$$q = (I - Q(D^2)^{-1} Q^T) a_{t+1} \qquad (7.33)$$

Analisando a última expressão, temos teoricamente $I - Q(D^2)^{-1} Q^T \neq 0$ desde que D(txt) é uma matriz diagonal com $D(i,i) = ||q_i|| = d_i^{1/2}$ onde q_i é o i-ésimo vetor coluna de Q . Mais ainda, considerando $A = Q\, D^{-1}\, D\, R$ sabemos que $Q' = Q\, D^{-1}$ e $R' = D\, R$; então $A = Q'\, R'$ onde Q' é uma matriz normalizada (i.e, $Q'^T Q' = I$).

Portanto, $I - Q(D^2)^{-1} Q^T = I - Q'\, Q'^T$, mas na seção (5.3) nós tivemos:

$$Q\, Q^T = Q_1\, Q_1^T + Q_2\, Q_2^T$$

e neste caso teremos:

$$I = \underline{Q}'^T\, \underline{Q}' = Q'\, Q'^T + Q_2'\, Q_2'^T$$

onde $\underline{Q}'^T = (Q', Q_2')$ e daí $Q_2'\, Q_2'^T = 0$.

Em resumo, Q_2' é uma matriz n x (n-t) com $Q_2'^T\, Q_2' = I$ e cujas colunas formam uma base ortogonal para o espaço nulo de vetores ortogonais às colunas de A .

Por outro lado, se n=t , Q_2' não existe e teremos:

$$Q'\, Q'^T = I \cong I = Q\, D^{2^{-1}}\, Q^T$$

mas observando (7.33) resulta que q=0 e $a_{t+1} = Qr$. Neste caso, a matriz $(Q\ a_{t+1}) \cong (Q\ Qr)$ é uma matriz de posto t .

Na seção (7.1) apresentou-se o critério seguido por DANIEL et.alii (3) que é similar ao exposto anteriormente com a diferença que no de Daniel não é garantido a ortogonalidade

$Q'^T q' = 0$ sendo preciso usar a reortogonalização para reduzir a instabilidade do processo.

No trabalho de ESCUDERO (9) , apresenta-se um algoritmo para atualizar a matriz no caso de adição de coluna.

Denotando $\bar{q}_{t+1} \cong q$ e $\bar{r}_{t+1} \cong \binom{r}{1}$ temos:

<u>Passo 1</u> : Os vetores coluna $\bar{q}_i$ e $\bar{r}_i$ para $i = 1,\ldots,t$ das matrizes $\bar{Q}$ e $\bar{R}$ são respectivamente q_i e r_i .

<u>Passo 2</u> : Obter vetor coluna $\bar{r}_{t+1}$ da expressão (7.32)

$$\bar{r}_{t+1,t+1} = 1$$

$$\bar{r}_{i,t+1} = \sum_{h=1}^{n} q_{hi} + a_{h,t+1} / d_i$$

$$\text{para } i = 1,\ldots, t$$

onde $d_i = D^2(i\ i) = ||q_i||_2^2$

<u>Passo 3</u> : Obter o vetor coluna $\bar{q}_{t+1}$ segundo (7.33)

$$\bar{q}_{h,t+1} = a_{h,t+1} - \sum_{i=1}^{t} q_{hi}\ \bar{r}_{i,t+1}$$

$$h = 1,\ldots, n$$

$$\bar{D}^2(i,i) = ||\bar{q}_{t+1}||_2^2$$

A atualização para o caso $n=t$ quase não é usada dado que em geral $n > t$.

Exemplo:

Seja $A = \begin{bmatrix} 1 & 1 & 1 \\ \varepsilon & 0 & 0 \\ 0 & \varepsilon & 0 \\ 0 & 0 & \varepsilon \end{bmatrix}$ cuja fatorização QR é:

$$A = \begin{bmatrix} 1 & 0 & 0 \\ \varepsilon & -\varepsilon & -\varepsilon \\ 0 & \varepsilon & 0 \\ 0 & 0 & \varepsilon \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ & 1 & 0 \\ & & 1 \\ & & \end{bmatrix}$$

Seja agora a coluna a_4 a ser adicionada i.e:

$$\bar{A} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ \varepsilon & 0 & 0 & 0 \\ 0 & \varepsilon & 0 & 0 \\ 0 & 0 & \varepsilon & \varepsilon \end{bmatrix}$$

Segundo a expressão (7.26):

$$\bar{A} = (A,\ a_{t+1})$$

$$\bar{A} = (Q,\ a_{t+1})\begin{pmatrix} R & 0 \\ & 1 \end{pmatrix}$$

$\bar{A} = \tilde{Q}\ \bar{R}$, temos:

$$\bar{A} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ \varepsilon & -\varepsilon & -\varepsilon & 0 \\ 0 & \varepsilon & 0 & 0 \\ 0 & 0 & \varepsilon & \varepsilon \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 0 \\ & 1 & 0 & 0 \\ & & 1 & 0 \\ & & & 1 \end{bmatrix}$$

$$\bar{A} = \qquad \tilde{Q} \qquad\qquad \bar{R}$$

Para encontrar as novas matrizes $\bar{Q}$ e $\bar{R}$ usaremos o algoritmo descrito.

<u>Passo 1</u> : Os vetores $\bar{q}_i$ e $\bar{r}_i$ para $i = 1,3$ das matrizes $\bar{Q}$ e $\bar{R}$ são as anteriores q_i e r_i i.e:

$$\bar{q}_1 = \begin{bmatrix} 1 \\ \varepsilon \\ 0 \\ 0 \end{bmatrix} \qquad \bar{q}_2 = \begin{bmatrix} 0 \\ -\varepsilon \\ \varepsilon \\ 0 \end{bmatrix} \qquad \bar{q}_3 = \begin{bmatrix} 0 \\ -\varepsilon \\ 0 \\ \varepsilon \end{bmatrix}$$

$$\bar{r}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \qquad \bar{r}_2 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \qquad \bar{r}_3 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$$

Passo 2 : Obter o vetor $\bar{r}_4 = \binom{r}{1}$ onde $\bar{r}_{44} = 1$

i = 1 $\quad \bar{r}_{14} = \sum_{h=1}^{4} q_{h1}\, a_{h4} \,/\, d_1$

$\bar{r}_{14} = 1$

i = 2 $\quad \bar{r}_{24} = \sum_{h=1}^{4} q_{h2}\, a_{h4} \,/\, d_2$

$\bar{r}_{24} = 0$

i = 3 $\quad \bar{r}_{34} = \sum_{h=1}^{4} q_{h3}\, a_{34} \,/\, d_3$

$\bar{r}_{34} = 1/2$

onde $d_i = ||q_i||_2^2$.

Passo 3 : Obter o vetor coluna $\bar{q}_4$

$$\bar{q}_{h4} = a_{h4} - \sum_{i=1}^{3} q_{hi}\, \bar{r}_{i4} \qquad h = 1,\dots,4$$

portanto:

$$\bar{q}_{14} = a_{14} - \sum_{i=1}^{3} q_{1i}\, \bar{r}_{i4} \quad \Rightarrow \quad \bar{q}_{14} = 0$$

$$\bar{q}_{24} = a_{24} - \sum_{i=1}^{3} q_{2i}\, \bar{r}_{i4} \quad \Rightarrow \quad \bar{q}_{24} = -\varepsilon/2$$

$$\bar{q}_{34} = a_{34} - \sum_{i=1}^{3} q_{3i}\, \bar{r}_{i4} \quad => \quad \bar{q}_{34} = 0$$

$$\bar{q}_{44} = a_{44} - \sum_{i=1}^{3} q_{4i}\, \bar{r}_{i4} \quad => \quad \bar{q}_{44} = \varepsilon/2$$

Sendo assim as novas matrizes $\bar{Q}\ \bar{R}$ são:

$$\bar{A} = \bar{Q}\ \bar{R} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \varepsilon & -\varepsilon & -\varepsilon & -\varepsilon/2 \\ 0 & \varepsilon & 0 & 0 \\ 0 & 0 & \varepsilon & \varepsilon/2 \end{bmatrix} \quad \begin{bmatrix} 1 & 1 & 1 & 1 \\ & 1 & 0 & 0 \\ & & 1 & 1/2 \\ & & & 1 \end{bmatrix}$$

7.4. <u>Apagar um vetor coluna da matriz A</u>

Dada a matriz A e sua decomposição QR, deseja-se apagar uma coluna da matriz A, obtendo a modificada $\bar{A}$ i.e:

$$A = (A_1 \;\; a \;\; A_2) = Q(R_1 \;\; r \;\; R_2)$$

$$\bar{A} = (A_1 \;\; A_2) = Q(R_1 \;\; R_2)$$

onde $\hat{R}$ é uma matriz de Hessemberg superior $(tx(t-1))$ com uma subdiagonal unitária desde a linha $p+1$ até t e tal que o vetor coluna r é a p-ésima coluna da matriz R. Por exemplo seja $t=6$ e $p=3$, então $\hat{R}$ toma a seguinte forma:

$$\hat{R} = \begin{array}{c} \begin{array}{ccccc} R_1 & & & R_2 & \end{array} \\ \begin{bmatrix} x & x & x & x & x \\ & x & x & x & x \\ & & x & x & x \\ & & x & x & x \\ & & & x & x \\ & & & & x \end{bmatrix} \end{array}$$

Observe-se que se $p=t$ então R_2 desaparece e $\hat{R} = R_1$ é já uma matriz triangular superior com diagonal unitária. Portanto para transformar $\hat{R}$ numa triangular superior, usa-se Matrizes de Givens e $\bar{A}$ será:

$$\bar{A} = Q\ P^T\ P\ \hat{R} \tag{7.34}$$

onde P é o produto de matrizes de Givens:

$$P = P_t^{t-1} \ldots P_{p+2}^{p+1}\ P_{p+1}^{p}$$

$$P^T = P_{p+1}^{p}\ P_{p+2}^{p+1} \ldots P_t^{t-1} \tag{7.35}$$

Desde que Q foi calculada pelo algoritmo NNQA, não é normalizada, e:

$$A = Q\ D^{-1}\ D\ R = Q\ D^{-1}\ R' \tag{7.36}$$

então:

$$\bar{A} = Q\ D^{-1}\ P^T\ P\ \hat{R}' \tag{7.37}$$

onde:

$$\hat{R}' = (R_1'\ R_2') \tag{7.38}$$

é tal que R_1' toma as primeiras $(p-1)$ colunas de DR e R_2' toma as últimas $(t-p)$ colunas de DR.

Aplicando P à $\hat{R}'$ obtemos:

$$\begin{bmatrix} \bar{R}' \\ 0 \end{bmatrix} = P\hat{R}'$$

isto é, conseguimos obter uma matriz triangular superior

$\bar{R}'(t-1)x(t-1)$ cuja diagonal não é necessariamente a identidade e 0 é o vetor linha (t-1).

Por outro lado admitindo que:

$$\bar{D}(i,i) = \bar{r}'_{ii} \tag{7.39}$$

resulta que:

$$\binom{\bar{R}'}{0} = \bar{D}\binom{\bar{R}}{0} \tag{7.40}$$

onde $\bar{R}$ é triangular superior $(t-1)x(t-1)$ com diagonal unitária.

Denotando:

$$(\bar{Q}' \ \bar{q}') \cong Q \ D^{-1} \ P^T \tag{7.41}$$

resulta:

$$\bar{A} = Q \ D^{-1} \ P^T \ \bar{D} \binom{\bar{R}}{0}$$

$$\bar{A} = (\bar{Q}' \ \bar{q}') \ \bar{D} \binom{\bar{R}}{0}$$

onde $(\bar{Q}', \bar{q}')$ é uma matriz ortonormal (nxt) ; pode ser provado que:

$$(\bar{Q}' \ \bar{q}')^T (\bar{Q}' \ \bar{q}) = I$$

$$P \ D^{-1} \ Q^T \ Q \ D^{-1} \ P^T = I$$

$$P \ D^{-1} \ D^2 \ D^{-1} \ P^T = P \ P^T = I$$

portanto $\bar{Q}'$ é uma matriz ortonormal $nx(t-1)$. Usando as expressões (7.40) e (7.41) temos:

$$\bar{A} = (\bar{Q}', \bar{q}')\binom{\bar{R}'}{0} = \bar{Q}'\ \bar{R}' \qquad (7.42)$$

Dado que $||\bar{q}_i||_2$ está já disponível, isto é, $\bar{D}(i,i)$ para $i = 1,\ldots, t-1$ resulta:

$$\bar{A} = \bar{Q}'\ \bar{R}' = \bar{Q}\ \bar{D}^{-1}\ \bar{D}\ \bar{R} = \bar{Q}\ \bar{R} \qquad (7.43)$$

onde $\bar{Q}$ é uma matriz ortogonal $(nx(t-1))$ e $\bar{R}$ é uma matriz triangular superior $(t-1)x(t-1)$ com diagonal unitária.

O procedimento computacional, considera o p-ésimo vetor coluna que será apagado da matriz A.

<u>Passo 0</u> : Se $p=t$ então faça $\bar{R} = R_1$.
Apague o vetor coluna q_t de Q faça $\bar{Q} = Q$ e finalize o procedimento.

<u>Passo 1</u> : Obter a matriz $\hat{R}' = (D\ R_1,\ D\ R_2)$.
Obter $\hat{R}_i'$ para $i = 1,\ldots,p-1$ onde $\hat{R}_i'$ é a i-ésima linha em $\hat{R}'$.

$\hat{r}_{ij}' = r_{ij}\ d_i^{1/2}$ para $j = 1,\ldots,p-1,p+1,\ldots t$

Obter $\hat{R}_p'$ i.e a p-ésima linha da matriz $\hat{R}'$

$\hat{r}_{pj}' = r_{pj}\ d_p^{1/2}$ para $j = p+1,\ldots, t$

Obter $\hat{R}_i'$ para $i = p+1,\ldots,1$

$\hat{r}_{ij}' = r_{ij}\ d_i^{1/2}$ para $j = i,\ldots, t$

fazer $\hat{r}_i' = \hat{r}_{i+1}'$ para $i = p,\ldots,t-1$ onde $\hat{r}_i'$ é a i-ésima coluna em $\hat{R}'$.

Passo 2 : Deve-se obter as matrizes $\bar{Q}$ e $\bar{R}$.

Primeiramente obter $\bar{R}'$ tal que se mantenha $P\hat{R}' = \binom{\bar{R}'}{0}$ e $P = P_t^{t+1} \ldots P_{p+2}^{p+1} P_{p+1}^{p}$ (Ver seção 4.3).

Mas observa-se que na matriz $\hat{R}$ temos $\hat{r}_{ii} = 1$ para $i = 1,2,\ldots,p-1$ e os elementos subdiagonais são:

$\hat{r}_{i+1,i} = 1$ para $i = p \ldots t-1$

Obter $\bar{D}(i,i) = \bar{r}'_{ii}$ para $i = 1,\ldots t-1$ ver (7.39)

Obter $\bar{R}$ tal que $\binom{\bar{R}'}{0} = \bar{D}\binom{\bar{R}}{0}$ mantenha-se.

Obter $\bar{R}_i$ para $i = 1,\ldots t-1$ onde $\bar{R}_i$ é a i-ésima linha de $\bar{R}$:

$\bar{r}_{ii} = 1$

$\bar{r}_{ij} = r'_{ij}/\bar{D}(i,i)$ para $j = i+1,\ldots t$

Obter a matriz $\bar{Q}'$ tal que $(\bar{Q}', \bar{q}') = Q\, D^{-1}\, P^T$ se mantenha.

Então, obter $Q' = Q\, D^{-1}$

$q'_{hi} = q_{hi}/d_i^{1/2}$ para $h = 1,\ldots n$

$i = 1,\ldots t$

Obter $\bar{Q}' = Q'\, P^T = Q'\, P_{p+1}^{p}\, P_{p+2}^{p+1} \ldots P_t^{t-1}$

Aqui deve-se cuidar os escalares c_k e s_k relacionados as matrizes de Givens P_{k+1}^{k} para $k = p,\ldots t-1$ que devem ser guardados enquanto obtém-se $\bar{R}'$.

Por último obter $\bar{Q}$ tal que se tenha (7.43) i.e $\bar{A} = \bar{Q}'\, \bar{R}' = \bar{Q}\, \bar{D}^{-1}\, \bar{D}\, \bar{R} = \bar{Q}\, \bar{R}$.

Uma outra maneira de calcular $\bar{Q}$ e $\bar{R}$, é apresentada por ESCUDERO (9) e consiste em obter $\bar{R}$ e $\bar{Q}$ simultaneamente de modo que não é necessário armazenar c_k e s_k .

7.5. Adição de um vetor linha à matriz A

Seja A fatorizada em QR com as mesmas características já descritas Queremos agora adicionar uma linha A_{n+1} à matriz A , seja por exemplo a última linha. A nova matriz modificada será:

$$\bar{A} = \begin{pmatrix} A \\ A_{n+1} \end{pmatrix} = \begin{pmatrix} Q & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} R \\ A_{n+1} \end{pmatrix} \equiv (\tilde{Q}\ e_{n+1})\ \hat{R} \tag{7.44}$$

onde $(\tilde{Q}\ e_{n+1})$ é uma matriz ortogonal $(n+1)x(t+1)$ tal que $\tilde{D}(i,i) = D(i,i)$ para $i = 1,\ldots t$ e $\hat{R}$ é uma matriz $(t+1)xt$ triangular superior com um vetor linha anexado.

Portanto, o objetivo é transformar $\hat{R}$ numa matriz triangular superior e isso será feito usando as Matrizes de Givens, isto é:

$$\bar{A} = (\tilde{Q}\ e_{n+1})\ P^T\ P\ \hat{R} \tag{7.45}$$

onde P é uma seqüência de Matrizes de Givens:

$$\begin{aligned} P &= P^t_{t+1} \cdots P^2_{t+1}\ P^1_{t+1} \\ P^T &= P^1_{t+1}\ P^2_{t+1} \cdots P^t_{t+1} \end{aligned} \tag{7.46}$$

O procedimento para obter as matrizes $\bar{Q}$, $\bar{R}$ será explicado posteriormente; primeiro analisaremos a matriz modificada $\bar{A}$:

$$\begin{aligned} \bar{A} &= \begin{pmatrix} QD^{-1} & 0 \\ 0 & 1 \end{pmatrix} P^T\ P \begin{pmatrix} DR \\ A_{n+1} \end{pmatrix} \\ \bar{A} &= \begin{pmatrix} Q' & 0 \\ 0 & 1 \end{pmatrix} P^T\ P \begin{pmatrix} R' \\ A_{n+1} \end{pmatrix} \\ \bar{A} &= (\tilde{Q}', q')\ P^T\ P\ \hat{R}' \end{aligned} \tag{7.47}$$

$$\bar{A} = (\bar{Q}', \bar{q}')\binom{\bar{R}'}{0}$$

Dado que $(\tilde{Q}'\ q')\ P^T = (\bar{Q}'\ \bar{q}')$, $P\ \hat{R}' = \binom{\bar{R}'}{0}$ sendo:

$$\bar{A} = \bar{Q}'\ \bar{R}' = \bar{Q}\ \bar{D}^{-1}\ \bar{D}\ \bar{R} = \bar{Q}\ \bar{R} \tag{7.48}$$

onde $\bar{Q}$ é uma matriz ortogonal $(n+1)xt$, $\bar{R}$ é uma matriz triangular superior (txt) com diagonal unitária e $\bar{D}$ é uma matriz diagonal (txt) tal que:

$$\bar{D}(ii) = \bar{r}'_{ii} \quad \text{para} \quad i = 1,\ldots t$$

O procedimento computacional é o seguinte:

<u>Passo 1</u> : Obter a matriz $R' = DR$ (ver 7.48)

$$r_{ij} = r_{ij}\ d_i^{1/2} \quad \text{para} \quad j = 1,\ldots t \quad e \quad i = 1,\ldots,t$$

<u>Passo 2</u> : Obter a matriz $Q' = Q\ D^{-1}$ (ver(7.48))

$$q_{hi} = q_{hi}\ /\ d_i^{1/2} \quad \text{para} \quad h = 1,\ldots n$$
$$i = 1,\ldots t$$

<u>Passo 3</u> : Fazer $q' = \binom{0}{1}$ onde 0 é um vetor coluna $(nx1)$, e

$$q'_{t+1,i} = 0 \quad \text{para} \quad i = 1,\ldots t$$

<u>Passo 4</u> : Obter as matrizes $\bar{R}'$ e $\bar{Q}'$, isto é obter os vetores colunas $\bar{r}'_k$ e $\bar{q}'_k$ para $k = 1,\ldots,t$ afim de conseguir $\hat{R}' \to \bar{R}'$.

Obter $P^k_{t+1}\ r_k$, isto é, calcular r_{hk} para $h = k,\ t+1$ como segue:

$$a = (r^2_{kk} + r^2_{t+1,k})^{1/2}$$

$$c = r_{kk}/a$$

$$s = r_{t+1,k}/a$$

$r_{kk} = a$

$r_{t+1,k} = 0$

$y = s/(1+c)$

Obter $P^k_{t+1}\ r_i$ para $i = k+1, \ldots\ t$, isto é, calcular r_{hi} para $h = k,\ t+1$.

$a = c\ r_{ki} + s\ r_{t+1,i}$

$r_{t+1,i} = y(r_{ki} + a) - r_{t+1,i}$

$r_{ki} = a$

Obter $(\bar{Q}'\ q')\ P^k_{t+1}$, isto é, calcular $q_{hk}\ P^k_{t+1}$ e $q_{h,t+1}\ P^k_{t+1}$ para $h = 1, \ldots\ n+1$

$a = c\ q_{hk} + s\ q_{h,t+1}$

$q_{h,t+1} = y(q_{hk} + a) - q_{h,t+1}$ (não é necessário para $k = \ell$)

$q_{hk} = a$

<u>Passo 5</u> : Obter matrizes $\bar{R}$ e $\bar{Q}$, ou seja, calcular o vetor linha $\bar{R}_i$ e o vetor coluna $\bar{q}_i$ para $i = 1, \ldots\ t$ tal que:

$r_{ij} = r_{ij}/r_{ii}$ para $j = 1+1, \ldots\ t$

$q_{hi} = q_{hi}\ r_{ii}$ para $h = 1, \ldots\ n+1$

$D^2(ii) = r^2_{ii}$

$r_{ii} = 1$

7.6. Apagar um vetor linha da matriz A

Considerando a matriz $A = QR$ com as mesmas características anteriores, sendo $n > t$, agora será a linha A_n de A a ser apagada. Assumiremos que este vetor está na parte inferior da matriz A e, se não for, pode-se permutar as linhas em A e Q a fim de que:

$$\binom{\bar{A}}{A_n} = \begin{pmatrix} \tilde{Q} & 0 \\ Q_n & 1 \end{pmatrix} \binom{R}{0} \cong (Q \; e_n) \binom{R}{0} \qquad (7.49)$$

onde $\tilde{Q}$ é uma matriz $(n\text{x}1)\text{xt}$ em que foi apagado o vetor linha Q_n da matriz $Q(n\text{xt})$ ortogonal, sendo que agora $\tilde{Q}$ não é ortogonal e (Q, e_n) também não o é:

$$\begin{bmatrix} \tilde{Q}^T & Q_n^T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{Q} & 0 \\ Q_n & 1 \end{bmatrix} = \begin{bmatrix} D^2 & Q_n^T \\ Q_n & 1 \end{bmatrix} \qquad (7.50)$$

e $Q_n \neq 0$ por definição.

Já é conhecido que $Q' = Q\,D^{-1}$ é uma matriz ortogonal e $R' = DR$. Portanto, o objetivo é transformar a $\tilde{Q}'$ numa matriz ortogonal afim de que:

$$\binom{\bar{A}}{A_n} = \begin{pmatrix} \bar{Q}' & 0 \\ 0 & 1 \end{pmatrix} \binom{\bar{R}'}{A_n} \qquad (7.51)$$

onde $\bar{Q}'$ e $\bar{R}'$ sejam os fatores de $\bar{A}$, a matriz $\bar{Q}'$ seja ortonormal e $\bar{R}'$ seja uma triangular superior.

O processo consiste em transformar

$$\begin{bmatrix} \tilde{Q}' & 0 \\ Q_n' & 1 \end{bmatrix} \qquad (7.52)$$

em ortonormal sem modificar a estrutura da matriz triangular superior R' de uma forma bastante similar ao que foi feito na seção (7.1) na linha (7.6). Isso significa que devemos obter os escalares σ e ρ, o vetor coluna $\tilde{q}((n-1)x1)$ e o vetor coluna $r(tx1)$ tais que a fatorização da matriz (7.52) possa ser escrita como:

$$\begin{bmatrix} \tilde{Q}' & 0 \\ Q'_n & 1 \end{bmatrix} = \begin{bmatrix} \tilde{Q}' & \tilde{q} \\ Q'_n & \sigma \end{bmatrix} \begin{bmatrix} I & r \\ & \rho \end{bmatrix} \tag{7.53}$$

sendo que a matriz modificada (7.49), fica:

$$\begin{bmatrix} \bar{A} \\ A_n \end{bmatrix} = \begin{bmatrix} \tilde{Q}' & \tilde{q} \\ Q'_n & \sigma \end{bmatrix} \begin{bmatrix} I & r \\ & \rho \end{bmatrix} \begin{bmatrix} R' \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} \bar{A} \\ A_n \end{bmatrix} = \begin{bmatrix} \tilde{Q}' & \tilde{q} \\ Q'_n & \sigma \end{bmatrix} \begin{bmatrix} R' \\ 0 \end{bmatrix} \tag{7.54}$$

Por outro lado, desde que a matriz:

$$\begin{bmatrix} \tilde{Q}' & \tilde{q} \\ Q'_n & \sigma \end{bmatrix} \tag{7.55}$$

é ortonormal, vai ser multiplicada por P^T (P é uma seqüência de Givens), portanto, a nova matriz é também ortonormal, e são as seguintes:

$$(Q'_n \quad \sigma)P^T = (0 \quad \pm 1) \tag{7.56}$$

$$(\tilde{Q}' \quad \tilde{q})P^T = (\bar{Q}' \quad \bar{q}') \tag{7.57}$$

ou

$$\begin{bmatrix} \bar{Q}' & \bar{q}' \\ 0 & \pm 1 \end{bmatrix} \tag{7.58}$$

Provaremos que a matriz (7.55) é ortonormal, e usaremos as expressões (7.56) e (7.57):

$$I = \begin{bmatrix} \tilde{Q}'^T & Q_n'^T \\ \tilde{q}^T & \sigma \end{bmatrix} \begin{bmatrix} \tilde{Q}' & \tilde{q} \\ Q_n' & \sigma \end{bmatrix}$$

$$I = P \begin{bmatrix} \tilde{Q}'^T & Q_n'^T \\ \tilde{q}^T & \sigma \end{bmatrix} \begin{bmatrix} \tilde{Q}' & \tilde{q} \\ Q_n' & \sigma \end{bmatrix} P^T$$

$$I = \begin{bmatrix} \bar{Q}'^T & 0 \\ \bar{q}'^T & \mp 1 \end{bmatrix} \begin{bmatrix} \bar{Q}' & \bar{q}' \\ 0 & \mp 1 \end{bmatrix}$$

$$I = \begin{bmatrix} \bar{Q}'^T \bar{Q}' & \bar{Q}'^T \bar{q}' \\ \bar{q}'^T \bar{Q}' & 1 \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & 1 \end{bmatrix}$$

Desde que (7.58) é ortonormal tem-se:

$$\left\| \begin{matrix} \bar{q}' \\ -+1 \end{matrix} \right\|_2^2 = 1 \qquad (7.59)$$

então $\bar{q}' = 0$; a expressão (7.57) pode também ser expressa por:

$$(\tilde{Q}' \ \tilde{q})P^T = (\bar{Q}' \ \bar{q}') = (\bar{Q}' \ \bar{0})$$

Por outro lado, seja $\begin{bmatrix} \bar{q}_i' \\ 0 \end{bmatrix}$ a i-ésima coluna da matriz $\begin{bmatrix} \bar{Q}' \\ 0 \end{bmatrix}$, temos:

$$1 = \left\| \begin{matrix} \bar{q}_i' \\ 0 \end{matrix} \right\|_2^2 \Rightarrow \left\| \bar{q}_i' \right\|_2^2 = 1$$

Usando todo o anterior, voltaremos a expressão (7.54) da matriz modificada usando Givens:

$$\begin{bmatrix} \bar{A} \\ A_n \end{bmatrix} = \begin{bmatrix} \tilde{Q}' & \tilde{q} \\ Q_n' & \sigma \end{bmatrix} P^T P \begin{bmatrix} R' \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} \bar{A} \\ A_n \end{bmatrix} = \begin{bmatrix} \bar{Q}' & 0 \\ 0 & \pm 1 \end{bmatrix} \begin{bmatrix} \bar{R}' \\ c \end{bmatrix} = \begin{bmatrix} \bar{Q}' & \bar{R}' \\ & c' \end{bmatrix} \tag{7.60}$$

onde c é um vetor linha (1xt) tal que:

$$\bar{A} = \bar{Q}' \; \bar{R}' = \bar{Q} \; \bar{D}^{-1} \; \bar{D} \; \bar{R} = \bar{Q} \; \bar{R} \quad ,$$

$\bar{D}$ é uma matriz diagonal (txt) tal que $\bar{D}(i,i) = \bar{r}'_{ii}$, $\bar{Q}$ é ortogonal ((n-1)xt) e $\bar{R}$ é triangular superior com diagonal unitária.

Trabalhando com a matriz (7.52) i.e:

$$\begin{bmatrix} \tilde{Q}' & 0 \\ Q_n' & 1 \end{bmatrix}$$

usaremos o procedimento exposto na seção (7.1) afim de encontrar os valores σ, ρ, r e $\tilde{q}$ com:

$$\begin{bmatrix} \tilde{Q}' & 0 \\ Q_n' & 1 \end{bmatrix} = \begin{bmatrix} \tilde{Q}' & \tilde{q} \\ Q_n' & \sigma \end{bmatrix} \begin{bmatrix} I & r \\ & \rho \end{bmatrix}$$

onde

$$Q^* = \begin{bmatrix} \tilde{Q}' & \tilde{q} \\ Q_n' & \sigma \end{bmatrix} = \begin{bmatrix} Q' & \tilde{q} \\ & \sigma \end{bmatrix}$$

é ortogonal e, denotando $Q' = \begin{bmatrix} \tilde{Q}' \\ Q'_n \end{bmatrix}$, temos o produto $Q^{*T}Q^{*}=I$:

$$Q^{*T}Q^{*} = \begin{bmatrix} Q'^T & \\ \tilde{q}^T & \sigma \end{bmatrix} \begin{bmatrix} Q' & \tilde{q} \\ & \sigma \end{bmatrix} = I$$

$$I = \begin{bmatrix} I & Q'^T\binom{\tilde{q}}{\sigma} \\ (\tilde{q}^T\sigma)Q' & ||\tilde{q}||_2^2 + \sigma^2 \end{bmatrix}$$

Portanto:

$$Q'^T\binom{\tilde{q}}{\sigma} = 0 \qquad (7.61)$$

e considerando agora a expressão (7.53) temos que:

$$\binom{0}{1} = Q'r + \binom{\tilde{q}}{\sigma}\rho \qquad (7.62)$$

Premultiplicando a expressão (7.62) por Q'^T e lembrando (7.61), obtemos:

$$Q'^T\binom{0}{1} = Q'^T\, Q'r\, Q'^T\binom{\tilde{q}}{\sigma}\rho$$

$$Q_n'^T = r \qquad (7.63)$$

Se $\rho \neq 0$, a matriz (7.52) é fatorizada com fatores QR dado que:

$$||\tilde{q}||_2^2 + \sigma^2 = 1$$

De outro modo, qualquer vetor $\tilde{q}$ que seja unitário e satisfaça a propriedade de ortogonalidade $\tilde{Q}'^T\,\tilde{q} = 0$ poderia fornecer a fatorização QR da matriz. Segundo a expressão:

$$\begin{bmatrix} 0 \\ 1 \end{bmatrix} = Q'r + \begin{bmatrix} \tilde{q} \\ \sigma \end{bmatrix} \rho$$

Se $\tilde{q}\ \rho = 0$ (7.52) não será uma matriz de posto coluna completo já que pela expressão anterior temos que a $(n+1)$-ésima coluna de (7.52) é uma combinação linear de suas n primeiras colunas (matriz Q').

Sendo assim, aquela matriz não terá a fatorização QR ; sua fatorização é chamada fatorização completa QR (exposta por GILL-MURRAY-WRIGHT (19)).

Nesta fatorização completa é necessário trocar as colunas mediante uma matriz de permutação P a fim de assegurar que as r colunas linearmente independentes sejam processadas primeiramente obtendo QAP de forma trapezoidal superior. Portanto o objetivo é agora transformar essa matriz numa triangular superior tal que:

$$QAP\ H_r \ldots H_1 = QAV = \begin{bmatrix} R & 0 \\ 0 & 0 \end{bmatrix}$$

Obteremos agora expressões para calcular σ e ρ ; partimos da expressão (7.62) e lembrando que $Q' = \binom{\tilde{Q}'}{Q'_n}$ e $Q'^{T}_{n} = r$ obtemos:

$$\begin{bmatrix} 0 \\ 1 \end{bmatrix} = Q'r + \begin{bmatrix} \tilde{q} \\ \sigma \end{bmatrix} \rho$$

$$\begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \tilde{Q}' \\ Q'_n \end{bmatrix} r + \begin{bmatrix} \tilde{q} \\ \sigma \end{bmatrix} \rho$$

$$\begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \tilde{Q}' \; Q_n'^T \\ Q'_n \; Q_n'^T \end{bmatrix} + \begin{bmatrix} \tilde{q} \\ \sigma \end{bmatrix} \rho$$

de onde:

$$\begin{bmatrix} 0 - \tilde{Q}' \; Q_n'^T \\ 1 - Q'_n \; Q_n'^T \end{bmatrix} = \begin{bmatrix} \tilde{q} \\ \sigma \end{bmatrix} \rho \tag{7.64}$$

Como (7.55) é ortogonal, $||\tilde{q}||_2^2 + \sigma^2 = 1$, e então

$$||\binom{\tilde{q}}{\sigma}\rho||_2^2 = \rho^2(||\tilde{q}||_2^2 + \sigma^2) = \rho^2$$

Portanto:

$$\rho^2 = ||\tilde{Q}' \; Q_n'^T||_2^2 + (1 - Q'_n \; Q_n'^T)^2$$

$$\rho^2 = Q'_n \; \tilde{Q}'^T \; \tilde{Q}' \; Q_n'^T + (1 - Q'_n \; Q_n'^T)^2 \tag{7.65}$$

Mais ainda, desde que $Q' = \binom{\tilde{Q}'}{Q'_n}$ é ortonormal, temos:

$$I = Q'^T \; Q' = (\tilde{Q}'^T \; Q_n'^T)\binom{\tilde{Q}'}{Q'_n}$$

$$I = \tilde{Q}'^T \; \tilde{Q}' + Q_n'^T \; Q'_n$$

$$\tilde{Q}'^T \; \tilde{Q}' = I - Q_n'^T \; Q'_n$$

Substituindo a última expressão em (7.65) obtemos:

$$\rho^2 = Q'_n(I - Q_n'^T \; Q'_n) \; Q_n'^T + (1 - Q'_n \; Q_n'^T)^2$$

$$\rho^2 = (Q'_n - Q'_n Q'^T_n Q'_n) Q'^T_n + (1 - Q'_n Q'^T_n)^2$$

$$\rho^2 = (Q'_n Q'^T_n - Q'_n Q'^T_n Q'_n Q'^T_n) + (1 - Q'_n Q'^T_n)^2$$

$$\rho^2 = Q'_n Q'^T_n (1 - Q'_n Q'^T_n) + (1 - Q'_n Q'^T_n)^2$$

$$\rho^2 = (Q'_n Q'^T_n + (1 - Q'_n Q'^T_n))(1 - Q'_n Q'^T_n)$$

Por fim:

$$\rho^2 = 1 - Q'_n Q'^T_n \qquad (7.65)$$

$$\rho^2 = 1 \sum_{i=1}^{t} q^2_{ni}/d_i \qquad (7.66)$$

Analisando a última parte da matriz (7.64) temos:

$$1 - Q'_n Q'^T_n = \sigma \rho$$

e então:

$$\rho^2 = \sigma \rho \qquad (7.67)$$

Portanto se $\rho \neq 0$ temos $\rho = \sigma$.

Se $\rho = 0$ então $Q'_n Q'^T_n = 1$ e como $Q'_n Q'^T_n + \sigma^2 \leq 1$ resulta em que $\sigma = 0$. Em qualquer caso $\rho = \sigma$.

Se ρ fornecida pelas expressões anteriores é "pequena", o procedimento pode tornar-se mal condicionado ao calcular $\tilde{q}$ e a propriedade de ortogonalidade de (7.52) pode ser perdida. Na realidade o valor de $\rho^2 = 1 - Q'_n Q'^T_n$ pode ser pequeno devido aos erros de arredondamento no cálculo de Q'_n i.é:

$$Q_n' = \sum_{i=1}^{t} q_{ni}/D_{(ii)}$$

e $D(ii) = ||q_i||_2$, e o uso de operações de raiz quadrada.

Porém, se usamos $\rho^2 = 1 - \sum_{i=1}^{t} q_{ni}/d_i$, as operações de raízes quadradas são evitadas e o risco de perder ortogonalidade é reduzido.

Resumindo tudo, temos:

$$r = Q_n'^T$$

$$\rho = \sigma$$

$$\rho^2 = 1 - Q_n' \, Q_n'^T \qquad \text{(usando 7.65)}$$

$$\tilde{q}_h = -\left(\sum_{i=1}^{t} q_{hi} \, q_{ni}/d_i\right)/\rho \quad \text{para} \quad h = 1,\ldots, n \qquad (7.68)$$

A expressão $Q_n' \, Q_n'^T + \sigma^2 \leq 1$ será justificada considerando $\underline{Q'}$ uma matriz ortonormal tal que:

$$\underline{Q'} \begin{bmatrix} \tilde{Q}' & 0 \\ Q_n' & 1 \end{bmatrix} = \begin{bmatrix} \underline{R'} \\ 0 \end{bmatrix}$$

onde:

$$\underline{R'} \cong \begin{bmatrix} I & r \\ & \rho \end{bmatrix}$$

Mais ainda a matriz $\underline{Q'}^T$ pode ser escrita como:

$$\underline{Q'}^T = (\underline{Q}_1' \; \underline{Q}_2')$$

onde $\underline{Q}_1'$ é a matriz $\begin{pmatrix} \tilde{Q}' & \tilde{q} \\ Q_n' & \sigma \end{pmatrix}$, já que $\begin{pmatrix} \tilde{Q}' & 0 \\ Q_n' & 1 \end{pmatrix}$ é uma matriz de posto coluna completo, então:

$$I = \underline{Q'}^T\, Q' = (Q_1'\ Q_2') \begin{bmatrix} Q_1'^T \\ Q_2'^T \end{bmatrix}$$

$$I = Q_1'\, Q_1'^T + Q_2'\, Q_2'^T$$

Denotando Q_n' como a n-ésima linha da matriz $\underline{Q'}^T$, obtemos:

$$1 = \underline{Q}_n'\, \underline{Q}_n'^T = \underline{Q}_{1_n}'\ \underline{Q}_{1_n}'^T + \underline{Q}_{2_n}'\ \underline{Q}_{2_n}'^T ,$$

sendo que:

$$0 \leq \underline{Q}_{1_n}'\ \underline{Q}_{1_n}'^T \leq 1$$

ou equivalentemente:

$$(Q_n'\ \sigma)\begin{pmatrix} Q_n'^T \\ \sigma \end{pmatrix} \leq 1 .$$

Se $n = t$, $Q'\,Q'^T = I$ e dado que $A = Q_1'R'$ onde $Q_1' = Q'$, a matriz Q_2' não existe.

O procedimento computacional é o seguinte: assumindo que A é uma matriz (nxt) onde $n > t$ e é de posto coluna completo:

Passo 1 : Obter o escalar ρ (7.66) tal que:

$$\rho^2 = 1 - \sum_{i=1}^{t} q_{ni}^2/d_i$$

$$\rho = (\rho^2)^{1/2}$$

Passo 2 : Obter o vetor coluna $\tilde{q}(n-1)x1$ (7.68) tal que:

$$q_h = -\left(\sum_{i=1}^{t} q_{hi}\, q_{ni}/d_i\right)/\rho$$

para $h = 1, \ldots, n-1$

Passo 3 : Obter a matriz $R' = DR$

$r_{ij} = r_{ij}\, d_i^{1/2}$ para $j = 1, \ldots t$, $i = 1, \ldots, t$

Obter a matriz $Q' = QD^{-1}$

$q_{hi} = q_{hi}/d_i^{1/2}$ para $h = 1, \ldots, n$, $i = 1, \ldots, t$

Passo 4 : Obter as matrizes $\overline{Q}'$ e $\overline{R}'$ isto é calcular o vetor coluna $\overline{q}_k'$ e o vetor linha $\overline{R}_k'$ para $k = t, \ldots, 1$ usando a matriz P .

Obter a atualização de ρ , reduzindo a zero o k-ésimo elemento de Q_n' tal que $(Q_n'\ \rho)P_k^{t+1}$ (7.56) seja realizada:

$\rho^2 = q_{nk}^2 + \rho^2$

$a = (\rho^2)^{1/2}$ se $\rho > 0$ então $a = -a$.

$c = \rho/a$

$s = q_{nk}/a$ $\qquad -1 < c < 0$

$\rho = a \qquad q_{nk} = 0$

$y = s/(1+c)$

Obter $(\overline{Q}'\ \overline{q}')\ P_k^{t+1}$ (7.57) isto é, calcular $\overline{q}_k'$ e $\overline{q}'$. Na realidade é calcular q_{hk} e q_h para $h = 1, \ldots, n-1$.

$a = c\, q_h + s\, q_{hk}$

$q_{hk} = y(q_h + a) - q_{hk}$

$q_h = a$

Obter $P_k^{t+1} R'$ i.e. calcular $P_k^{t+1} R_i'$ para $i = k, \ldots, t$

Obter r_{kk} e c_k, onde c_k é o k-ésimo elemento do vetor linha c.

$c_k = sr_{kk}$

$r_{kk} = -cr_{kk} > 0$

Obter r_{ki} e c_i para $i = k+1, \ldots, t$

$a = -cr_{ki} + sc_i$

$c_i = sr_{ki} + cc_i$

$r_{ki} = a$

Passo 5 : Obter matrizes $\bar{R}$, $\bar{Q}$ e $\bar{D}$

$r_{ij} = r_{ij}/r_{ii}$ para $i = i, \ldots, t$

$j = i+1, \ldots, t$

$q_{hi} = q_{hi}\ r_{ii}$ para $h = 1, \ldots, n-1$

$D^2(ii) = r_{ii}^2$

$r_{ii} = 1$

Afim de evitar erros de arredondamento computacionais e sabendo que:

$\bar{D}^2(k,k) \equiv \bar{d}_k = \bar{r}'^2_{kk}$ para $k = 1, \ldots, t$ os cálculos anteriores de r_{kk} são substituidos por:

$c_k = s\ d_k^{1/2}$

$\bar{D}^2(kk) = c^2\ d_k > 0$,

com os vetores $\bar{q}_k$ e $\bar{R}_k$ para $k = 1,\ldots, t$, são calculados no passo 5 como:

$$\bar{r}_{ki} = \bar{r}_{ki}/|\bar{d}_k|^{1/2} \quad \text{para} \quad i = k+1,\ldots, t$$

$$\bar{r}_{kk} = 1$$

$$\bar{q}_{hk} = \bar{q}_{hk}|\bar{d}_k|^{1/2}$$

Estas modificações reduzem o uso futuro de operações de raízes quadradas.

CAPÍTULO VIII

CÁLCULO DOS ESTIMADORES DOS MULTIPLICADORES DE LAGRANGE PARA PROBLEMAS NÃO LINEARES RESTRITOS LINEARMENTE

8.1. Motivação: Estratégia dos conjuntos ativos em PNL com restrições lineares

Os multiplicadores de Lagrange, desempenham um papel muito importante na maioria dos algoritmos eficientes para otimização restrita, principalmente nos algoritmos que seguem a metodologia de Estratégia dos Conjuntos Ativos.

Antes de apresentar os aspectos teóricos do cálculo destes multiplicadores e/ou seus estimadores serão expostos suas utilizações e uma visão geral da metodologia da Estratégia dos Conjuntos Ativos.

8.1.1. Enunciado do Problema

Seja o problema de otimização restrita não linear formulado como segue:

$$\text{(P1)} \quad \begin{array}{l} \text{Min} \quad F(x) \\ x \in \mathbb{R}^n \\ \text{s.a.} \\ \qquad A^T x \geq b \end{array} \tag{8.1}$$

sendo A^T a matriz (mxn), x o vetor (nx1), b o vetor (mx1).

Um ponto $\overset{*}{x}$ é um <u>mínimo local forte</u> de F(x) se F(x) é definido numa vizinhança de radio δ do ponto $\overset{*}{x}$ e existe um $\varepsilon (0 < \varepsilon \leq \delta)$ tal que $F(\overset{*}{x}) < F(x)$ / $||x-\overset{*}{x}|| < \varepsilon$.

Um ponto $\overset{*}{x}$ é um <u>mínimo global</u> de uma função F(x) se $\forall x \in E^n$ cumpre $F(\overset{*}{x}) \leq F(x)$.

As notações a serem usadas são: a_i^T - i-ésima linha da matriz A, $g(x) = \nabla F(x)$, $G(x) = \nabla^2 F(x)$.

8.1.2. <u>Condições de otimalidade para o problema P1</u>

Para que um ponto $\overset{*}{x}$ viável, seja solução do problema P1, deve satisfazer certas condições de otimalidade. Antes de serem mencionadas definiremos alguns conceitos.

Considerando um ponto $\bar{x}$, a i-ésima restrição: $a_i^t \, x \geq b_i$ diz-se:

1) Ativa : se $a_i^T \bar{x} = b_i$

2) Inativa : se $a_i^T \bar{x} > b_i$

3) Satisfeita : se a restrição é ativa ou inativa, neste caso $\bar{x}$ é viável com respeito à i-ésima restrição.

4) Violada : se $a_i^T \bar{x} < b_i$, neste caso $\bar{x}$ é dito inviável (em particular, com respeito à i-ésima restrição).

Definimos:

1) $\hat{A}$ a matriz de restrições ativas i.e. cujas colunas são os vetores $\{a_i\}$ correspondentes às restrições ativas no ponto $\overset{*}{x}$.

2) t é o número de restrições ativas.

3) Z é uma matriz cujas colunas formam uma base para o subes-