DECOMPOSIÇÃO QR

EP3 - MAC0300

VICTOR SENA MOLERO (8941317)

November 22, 2015

CONTENTS

ABSTRACT

Esse é o relatório sobre o terceiro EP de MAC0300, que tem como objetivo a implementação da Decomposição QR e a resolução do problema dos quadrados mínimos. Os dois conceitos serão revisados brevemente e os principais problemas sobre a implementação serão discutidos com mais cuidado.

1 INTRODUÇÃO

Além das explicações sobre os tópicos abordados naturalmente pelo EP, este relatório justifica as decisões feitas durante a implementação do algoritmo e mostra como foram criados os métodos para geração de testes para o programa.

2 O PROBLEMA DOS QUADRADOS MÍNIMOS

Com $n \leqslant m$, ada uma matriz $A \in \mathbb{R}^{n \times m}$ e um vetor $b \in \mathbb{R}^m$ encontrar $x \in \mathbb{R}^n$ que minimize $||b - Ax||_2$.
Isto é equivalente a dizer que dado um sistema sobredeterminado, de m equações com n variáveis onde $n \leqslant m$, deseja-se encontrar os valores das n variáveis que minimizem o erro entre aplicar as equações a essas variáveis e os resultados dados pelo problema.
Uma aplicação clara disso é, dada uma base de polinômios de dimensão n, interpolar o polinômio que melhor aproxima alguns pontos dados da função.
O problema pode ter uma única ou infinitas soluções. Dependendo da singularidade de A. Nosso objetivo será encontrar uma.

2.1 Decomposição QR

A decomposição QR encontra, para uma matriz $A \in \mathbb{R}^{n \times m}$, duas matrizes, uma $Q \in \mathbb{R}^{n \times m}$ e uma $R \in \mathbb{R}^{m \times m}$ tais que $A = QR$, Q é ortogonal e R é triangular superior.
Para isso, basta escolher inteligentemente uma série de refletores (ou rotatores) Q_i^T na matriz A de forma a transformá-la em uma R triangular superior. Assim, teremos que, se Q^T é o produto de todas as Q_i^T aplicadas, $Q^T A = R$. Já que Q^T é produto de refletores, que são ortogonais, é ortogonal, logo $Q^{-T} = Q$ e $A = QR$, assim obtemos a decomposição QR.

2.2 Usando a Decomposição

Para resolver o problema dos quadrados mínimos usando QR lembramos que devemos achar x tal que $b - Ax$ tem norma 2 mínima. Porém, se premultiplicarmos a expressão por Q^T teremos:

$$Q^T(b - Ax) = Q^T b - Q^T Ax = Q^T b - Q^T QRx = Q^T b - Rx$$

Nós conseguimos encontrar x tal que $Q^T b = Rx$ com substituição para trás, já que R é triangular superior, assim, nós temos uma solução para um problema dos quadrados mínimos de $\hat{A}$ e $\hat{b}$ onde $\hat{A} = Q^T A$ e $\hat{b} = Q^T b$. Mas já que Q é ortogonal, $||b - Ax||_2 = ||Q^T(b - Ax)||_2$. Ou seja, ao resolver este problema, resolvemos nosso problema inicial, já que o x que minimiza um também minimiza o outro.

3 IMPLEMENTAÇÃO E PROBLEMAS

Em sala de aula foram discutidas algumas dúvidas quanto à implementação e sobre práticas que minimizariam a quantidade de operações ou espaço ou que maximizariam a precisão da solução. O fato é que se A tem sempre posto completo, só existe uma opção a ser feita quando a memória ou tempo. Já se A tem posto incompleto, devemos aplicar o pivoteamento de colunas e existe mais uma forma de implementar isso e de calcular qual coluna deve ser pivotada. Vamos discutir um problema por vez.

3.1 Aplicando a Reflexão

É comentado no livro o fato de que existem várias maneiras de aplicar a reflexão, mas o custo de aplicar muda drásticamente. Além disso, se formos aplicar a reflexão orientada a linhas, devemos usar um vetor auxiliar.
No meu EP, eu aproveitei o espaço ainda livre no vetor γ para realizar os calculos necessários. O vetor em questão, no passo k, tem k posições calculadas (incluindo a que é calculada no passo atual). Já que o vetor é inicializado com m espaços, ele tem, no passo k, $m - k$ espaços livres. Esta quantidade é exatamente a usada pelo vetor auxiliar mencionado no livro.

3.2 Escalamento

Para evitar overflows nos cálculos das normas das colunas, é necessário normaliza-las de alguma maneira. Existem alguns jeitos de fazer isso e o livro menciona dois.
Inicialmente, falando de problemas com posto completo, o livro fala sobre dividir cada coluna pelo seu máximo elemento, evitando overflow. Depois, para resolver problemas de posto incompleto, é comentado o método de dividir, à priori, a matriz inteira pelo maior elemento da matriz.
Eu resolvi utilizar o segundo método. Ele tem vantagens e desvantagens. Além de me parecer mais elegante do que as maneiras que consegui pensar em implementar o outro método, gasta menos memória do que os métodos que eu pensei e os que foram discutidos em aula.
Dividir a matriz toda por um só escalar funciona pois minimizar $b - A\alpha * (1/\alpha) * x$ é minimizar $b - Ax$, ou seja, se resolvermos um problema onde dividimos A por α, basta multiplicar todos os elementos da R obtida por α novamente e teremos a decomposição do problema original.

4 TESTES

Para testar o ep, eu criei exemplos de polinomios em bases arbritárias para testar a interpolção feita pela decomposição QR. Para gerar os testes que eu usei basta acessar a pasta *exemplos* e rodar *sh criaTestes.sh*, assim, os arquivos de teste serão criados na pasta *exemplos/gerador* e podem ser usados diretamente no programa main.
A imagem *exemplos/plots/poly3.png* é uma saída gráfica para o exemplo que é gerado em *exemplos/testes/poly3.dat*. Eu queria criar mais desses, mas tive trabalho para gerar as imagens e meu tempo está acabando.

4.1 Formato de entrada

Acho pertinente colocar um comentário sobre o formato de entrada de dados adotado no ep. São dados inicialmente os dois inteiros n e m, depois, $n * m$ linhas que representam a matriz de forma exatamente igual à adotada nos eps 1 e 2. Depois, ainda são inseridos os n elementos do vetor b.