

MÉTODOS NUMÉRICOS - LISTA DE EXERCÍCIOS I

SÉRGIO CORDEIRO

SUMÁRIO

1. Exercícios	2
2. Análise e conclusões	11
3. Anexos	12
Glossário	13
Siglas	13
Referências	14

1. EXERCÍCIOS

1. Desenvolva um algoritmo em C para realizar a multiplicação de duas matrizes. Não se esqueçam de verificar se a multiplicação é possível antes de realizá-la. Teste os resultados obtidos com aqueles gerados pelo MATLAB. Considerar precisão simples e precisão estendida. Para comparar os resultados calcule a norma 2 da matriz resultado. Comente!

O programa abaixo gera, no MATLAB, duas matrizes com valores aleatórios, grava os valores em disco, calcula o produto e a norma 2 da matriz-resultado.

LISTING 1. geramat.m

```
1 function geramat(n)
2 % Gera duas matrizes 'n' x 'n' a partir de valores aleatórios na faixa ]0 - 1[, grava-
   as em disco, calcula o produto e a norma 2 do resultado.
3 A = rand(n);
4 save('MatA', 'A');
5 B = rand(n);
6 save('MatB', 'B');
7 disp(sprintf("Matrizes geradas e gravadas."));
8 C = A * B;
9 disp(sprintf("Norma 2 do produto = %f", norm(C, 2)));
10 end
```

O programa **exercmat.c**, em anexo, escrito em C, lê as matrizes geradas pelo MATLAB, calcula o produto e a norma 2 da matriz-resultado para diversas precisões (32, 64 e 80 bits) e para alguns valores de n (rank das matrizes). Basta digitar:

```
exercmat 1 n
```

Os resultados obtidos estão tabelados a seguir.

n	precisão	origem	norma 2
3	dupla	MATLAB	1.782348
3	simples	C	1.793582
3	dupla	C	1.793582
3	estendida	C	1.793582
30	dupla	MATLAB	232.297907
30	simples	C	232.651062
30	dupla	C	232.651093
30	estendida	C	232.651093
300	dupla	MATLAB	22497.289165
300	simples	C	22501.390625
300	dupla	C	22501.443224
300	estendida	C	22501.443224

2. Desenvolva um algoritmo em C para resolver sistemas lineares triangulares (Superior e Inferior). Teste os resultados obtidos com aqueles gerados pelo MATLAB. Considerar precisão simples e precisão estendida. Para comparar os resultados calcule a norma 2 da matriz resultado. Comente!

O programa abaixo gera, no MATLAB, dois sistemas triangulares com valores aleatórios, um superior e outro inferior, grava os valores em disco e os resolve por meio do cálculo da inversa. Ele também calcula a norma 2 da matriz-coluna-resultado.

LISTING 2. `geramattri.m`

```

1 function geramattri(n)
2 % Gera dois sistemas triangulares 'n' x 'n', um superior e outro inferior, a partir de
  valores aleatórios na faixa ]0 - 1[, grava-os em disco, resolve-os e calcula a
  norma 2 do resultado.
3 A = B = zeros(n, n);
4 for m = 1:n
5     nvals = 1 + n - m;
6     A(m, m:n) = rand(1, nvals);
7     B(nvals, 1:nvals) = rand(1, nvals);
8 end
9 bA = rand(1, n)';
10 bB = rand(1, n)';
11 TS = [A bA];
12 TI = [B bB];
13 save('TS', 'TS');
14 save('TI', 'TI');
15 disp(sprintf("Sistemas gerados e gravados."));
16 INVA = A ^ (-1);
17 INVB = B ^ (-1);
18 xA = INVA * bA;
19 xB = INVB * bB;
20 disp(sprintf("Norma 2 dos resultados = %f e %f", norm(xA, 2), norm(xB, 2)));
21 end

```

O programa **exercmat.c**, em anexo, escrito em C, lê as matrizes geradas pelo MATLAB, resolve os sistemas e calcula a norma 2 da matriz-coluna-resultado para diversas precisões (32, 64 e 80 bits) e para alguns valores de n (rank das matrizes). Basta digitar:

exercmat 2 n

Os resultados obtidos estão tabelados a seguir.

n	precisão	origem	norma 2	
			TS	TI
3	dupla	MATLAB	7.698141	6.230213
3	simples	C	7.698142	6.230213
3	dupla	C	7.698141	6.230213
3	estendida	C	7.698141	6.230213
30	dupla	MATLAB	5223.534435	23824.246320
30	simples	C	5223.534668	23824.248047
30	dupla	C	5223.534435	23824.248557
30	estendida	C	5223.534435	23824.246320
300	dupla	MATLAB	1.075×10^{56}	2.105×10^{55} ¹
300	simples	C	<i>overflow</i>	<i>overflow</i>
300	dupla	C	1.075×10^{56}	2.105×10^{55} ¹
300	estendida	C	1.075×10^{56}	2.105×10^{55} ¹

¹Os valores são diferentes mas, como são muito grandes, a diferença não pôde ser mostrada, porque aparece apenas nos últimos dígitos.

3. Desenvolva um algoritmo em C para resolver sistemas lineares utilizando eliminação de Gauss com pivoteamento. Como subproduto calcule o determinante da matriz. Considerar precisão simples e precisão estendida. Para comparar os resultados calcule a norma 2 da matriz resultado. Comente!

O programa abaixo gera, no MATLAB, um sistema com valores aleatórios, grava-o em disco, resolve-o por meio da matriz inversa e calcula o determinante e a norma 2 da matriz-coluna-resultado.

LISTING 3. `gerasis.m`

```

1 function gerasis(n)
2 %Gera um sistema a partir de valores aleatórios na faixa ]0 - 1[, grava-o em disco,
   resolve-o e calcula o determinante e a norma 2 do resultado.
3 A = rand(n);
4 bA = rand(1, n)';
5 S = [A bA];
6 save('S', 'S');
7 disp(sprintf("Sistema gerado e gravado.));
8 disp(sprintf("Determinante = %f", det(A)));
9 INVA = A ^ (-1);
10 xA = INVA * bA;
11 disp(sprintf("Norma 2 do resultado = %f", norm(xA, 2)));
12 end

```

O programa **exercmat.c**, em anexo, escrito em C, lê o sistema gerado pelo MATLAB, resolve-o pela técnica de eliminação de Gauss com pivoteamento e calcula o determinante e a norma 2 da matriz-coluna-resultado para diversas precisões (32, 64 e 80 bits) e para alguns valores de n (rank das matrizes). Basta digitar:

exercmat 3 n

Os resultados obtidos estão tabelados a seguir.

n	precisão	origem	determinante	norma 2
3	dupla	MATLAB	-0.112721	4.464723
3	simples	C	-0.112721	4.464724
3	dupla	C	-0.112721	4.464723
3	estendida	C	-0.112721	4.464723
30	dupla	MATLAB	26.168290	1.938658
30	simples	C	26.168304	1.938658
30	dupla	C	26.168290	1.938658
30	estendida	C	26.168290	1.938658
300	dupla	MATLAB	2.694×10^{145} ²	25.915637
300	simples	C	<i>overflow</i>	25.915474
300	dupla	C	2.694×10^{145} ²	25.915637
300	estendida	C	2.694×10^{145} ²	25.915637

²Idem nota 1.

4. Apresente uma pesquisa sobre aplicação de sistemas lineares nas seguintes áreas: Eletromagnetismo, controle, SEP, modelagem de sistemas. Apresente no mínimo 4 aplicações por área, faça uma breve descrição de cada uma delas destacando as características das matrizes e os métodos utilizados para a solução do sistema.

A tabela abaixo apresenta as aplicações de sistemas lineares, as características das matrizes e os métodos usados para solução, considerando as áreas citadas.

Eletromagnetismo

Aplicação	Características das Matrizes	Métodos de Solução	
		Problema	Sistemas
Análise de sinais de radar	Densas Simétricas Complexas Não-Hermitianas	Método dos Momentos	Eliminação Gaussiana [DONGARRA 2010]
Radiação e espalhamento	Densas	Método dos Momentos	Interpoladores de Lagrange Precondicionadores Métodos Iterativos [GIBSON 2015, RYLANDER 2013]
Solução das Equações de Maxwell no Domínio do Tempo	Grandes	Método das Diferenças Finitas	Métodos Iterativos [GIBSON 2015]
Cálculo de correntes parasitas	Grandes Esparsas Simétricas Matriz em banda Definidas-positivas	Método dos Elementos Finitos	Decomposição espectral [GIBSON 2015, RYLANDER 2013]

SEP

Aplicação	Características das Matrizes	Métodos de Solução	
		Problema	Sistemas
Estimativa de estado harmônico	Sistemas subdeterminados Presença de ruído Observabilidade parcial	Inferência estatística Filtragem adaptativa Simulação de Monte Carlo Maximização de esparsidade	Mínimos quadrados Decomposição SVD [ZHOU 2009, YU 2005] [LIAO 2006, ARUMUGAM 2011]
Estimativa de estado transitório	Observabilidade parcial Sistemas não-lineares	Inferência estatística Linearização	Métodos Iterativos [YU 2005]
Análise de fluxo de potência	Sistemas subdeterminados Simétricas Definidas-positivas Matriz em banda	Matriz admitância (Ybus) Matriz impedância (Zbus)	Eliminação Gaussiana Gauss-Seidel [BENEDITO 2013]
Verificação de projeto de redes de distribuição	Esparsas Grandes	Diminuição de rank Técnicas <i>knapsack</i>	Métodos Iterativos [FERZLI 2007]

Controle

Aplicação	Características das Matrizes	Métodos de Solução	
		Problema	Sistemas
Solução das equações de estado	Densas Pequenas	Transformada Z	Inversão [PHILLIPS 1995]
Controle ótimo	Densas Pequenas	Mínimos Quadrados	Inversão [PHILLIPS 1995]
Projeto de filtros, controladores e observadores	Densas Pequenas	Transformada bilinear	Inversão [PHILLIPS 1995]
Identificação de sistemas	Densas Pequenas	Regressão polinomial	Inversão [PHILLIPS 1995]

Modelamento

Aplicação	Características das Matrizes	Métodos de Solução	
		Problema	Sistemas
Aproximação de funções	Densas Grandes	Análise de subespaços	Decomposição QR Decomposição SVD [OVERSCHEE 1996]
Classificação	Densas Grandes Valores discretos	Regressão logística SVM	Métodos Iterativos [LIN 2008]
Aglutinação	Densas Grandes	<i>k-means</i> Aglomeração hierárquica	Mínimos Quadrados [STEINBACH 2000]
Descoberta de parâmetros	Densas Simétricas	PCA	Decomposição SVD [MOORE 1981]

2. ANÁLISE E CONCLUSÕES

Os resultados expostos nas tabelas referentes aos exercícios 1 a 3 mostram que, em todos os casos, pode ser notada uma ligeira diferença entre a precisão simples e a dupla. Essa diferença se torna mais nítida à medida que o tamanho do problema aumenta. A precisão estendida, por outro lado, não parece trazer benefícios com relação à precisão dupla. Todos esses efeitos são previstos pela teoria.

Além da perda de precisão, em problemas maiores ocorre também o problema de *overflow*, quando o valor não pode ser representado com o número de bits disponível.

Não foram examinados os efeitos da utilização de precisão quádrupla ou óctupla, porque o *hardware* disponível para os testes não as suporta diretamente e, assim, as funções precisariam ser implementadas unicamente por meio de bibliotecas de *software*, o que tornaria a execução muito lenta. Pelo mesmo motivo, não foi investigada a precisão variável. A tabela abaixo apresenta o esforço computacional necessário para solução de cada problema, de acordo com o tamanho da entrada. Os números ajudam a avaliar o motivo dos erros de arredondamento. Percebe-se que a operação cujo custo cresce mais rapidamente com o tamanho de problema é a multiplicação de matrizes; para essa operação os erros devem também aumentar mais sensivelmente do que para as demais.

Problema	Número de operações		
	$m = 3$	$m = 30$	$m = 300$
Multiplicação de duas matrizes	27	27000	27000000
Solução de sistema triangular	21	1020	91200
Eliminação Gaussiana	48	19551	18137046
Cálculo do determinante	3	30	300

3. ANEXOS

Os seguintes arquivos constam do anexo (arquivo **exercmat1.zip**):

- arquivo fonte em C **exercmat.c**
 - arquivos fonte em MATLAB:
 - **geramat.m**: problema 1
 - **geramattri.m**: problema 2
 - **gerasis.m**: problema 3
 - arquivos de dados:
 - MatA_n e MatB_n : problema 1
 - TS_n e TI_n : problema 2
 - Sn : problema 3
-

GLOSSÁRIO

Principal Component Analysis: É uma técnica que transforma um conjunto de n observações $v_i \in \mathfrak{D}$, cada uma delas consistindo do valor medido de uma variável $v_j \in \mathfrak{B}$, em um conjunto de n observações $p_i \in \mathfrak{P}$, cada uma delas consistindo do valor referente a uma variável $u_j \in \mathfrak{U}$, de forma que o número de elementos de \mathfrak{U} seja menor que o número de variáveis originais e o mínimo de informação seja perdido no processo.

Support Vector Machines: É uma técnica de aprendizado de máquina supervisionado que aprende a classificar objetos com base em etiquetas a eles associadas, tornando-se então capaz de classificar objetos desprovidos de etiquetas.

k-means: É uma técnica de aglutinação que usa alguma média dos valores dos elementos como critério para particionamento.

Aglutinação: É a tarefa de particionar um conjunto de forma que os elementos de cada conjunto apresentam máxima similaridade, de acordo com algum critério predefinido. Também conhecida pelo termo em inglês *clustering*.

Matriz em banda: Também chamada de matriz diagonalmente dominante, é aquela matriz esparsa em que apenas elementos próximos à diagonal principal são não-nulos.

Técnicas knapsack: São técnicas para solução do problema alcunhado de *knapsack*, de selecionarem-se os itens mais valiosos de uma coleção.

SIGLAS

PCA: *Principal Component Analysis*.

SVM: *Support Vector Machines*.

REFERÊNCIAS

- [ARUMUGAM 2011] Ugasciny ARUMUGAM, Nursyarizal Mohd NOR e Mohd Faris ABDULLAH, **A Brief Review on Advances of Harmonic State Estimation Techniques in Power Systems**. Disponível em <http://www.ijiee.org/papers/34-I040.pdf>, acesso em 05/03/2016.
- [BENEDITO 2013] Raphael Augusto de Souza BENEDITO, **Fluxo de Potência (de Carga)**. Disponível em <http://paginapessoal.utfpr.edu.br/raphaelbenedito>, acesso em 05/03/2016.
- [DONGARRA 2010] Jack DONGARRA e Piotr LUSZCZEK, **Reducing the time to tune parallel dense linear algebra routines with partial execution and performance modelling**. Disponível em <http://www.netlib.org/lapack/lawnspdf/lawn235.pdf>, acesso em 05/03/2016.
- [FERZLI 2007] Imad A. Ferzli, Naser Farid e Lars Kruse, **Early power grid verification under circuit current uncertainties**, in Low Power Electronics and Design (ISLPED), 2007 ACM/IEEE International Symposium, 27-29 Aug. 2007, pp. 116 a 121.
- [GIBSON 2015] Walton GIBSON, **The Method of Moments in Electromagnetics**, CRC Press, 2015.
- [LIAO 2006] LIAO Huaiwei, **Power System Harmonic State Estimation and Observability Analysis via Sparsity Maximization**. Disponível em <https://www.andrew.cmu.edu/user/hliao/paper/liao-hse-pwrs.pdf>, acesso em 05/03/2016.
- [LIN 2008] LIN Chih-Jen, WENG Ruby C. e S. Sathiya KEERTHI, **Trust Region Newton Method for Logistic Regression**, in The Journal of Machine Learning Research, Volume 9, Sep. 2008, pp. 627 a 650.
- [MOORE 1981] B. MOORE, **Principal component analysis in linear systems: Controllability, observability, and model reduction**, in IEEE Transactions on Automatic Control, Volume 26, Issue 1, ISSN 0018-9286, pp. 17 a 32.
- [OVERSCHEE 1996] Peter van OVERSCHEE e Bart de MOOR, **Subspace Identification for Linear Systems**, Kluwer Academic Publishers, Boston, 1996, ISBN 13:978-1-4613-8061-0.
- [PHILLIPS 1995] Charles L. PHILLIPS e H. Troy NAGLE, **Digital Control System Analysis and Design**, Prentice Hall, Englewood Cliffs, 1995, 3rd Ed..
- [RYLANDER 2013] Thomas RYLANDER, Pär INGELSTRÖM e Anders BONDESON, **Computational Electromagnetics**, Springer, New York, 2013, 2nd Ed., ISBN 978-1-4614-5350-5: Chap. 1, pp. 1 a 3.
- [STEINBACH 2000] Michael STEINBACH, George KARYPIS e Vipin KUMAR, **A Comparison of Document Clustering Techniques**. Disponível em https://www.cs.cmu.edu/~dunja/KDDpapers/Steinbach_IR.pdf, acesso em 10/03/2016.
- [YU 2005] YU Kent, **Harmonic State Estimation and Transient State Estimation**. Disponível em http://ir.canterbury.ac.nz/bitstream/10092/1108/1/thesis_fulltext.pdf, acesso em 05/03/2016.
- [ZHOU 2009] ZHOU Niancheng, LIN Li e ZHU Jizhong, **An Approach to Harmonic State Estimation of Power System**. Disponível em <http://www.scirp.org/journal/PaperDownload.aspx?paperID=730>, acesso em 05/03/2016.
-

Programas testados com **Octave** 4.0.0 e **MinGW** C 4.8.2:

<https://www.gnu.org/software/octave/>

<https://www.mingw.org>

Texto formatado com **pdflatex** em ambiente **MiKTeX** 2.9:

<http://miktex.org/download/>