INSTITUTO FEDERAL DO ESPÍRITO SANTO

CAMPUS VITÓRIA

MESTRADO PROFISSIONAL EM TECNOLOGIAS SUSTENTÁVEIS

**ÁLVARO LUIZ LAGO DE MENEZES**

**RELATÓRIO DO LABORATÓRIO 5: PROBLEMA DE MINIMIZAÇÃO SEM RESTRIÇÕES**

**MÉTODOS DE OTIMIZAÇÃO: MATEMÁTICOS E HEURÍSTICOS**

Vitória

2019

SUMÁRIO

SUMÁRIO 2

1 INTRODUÇÃO 3

2 DESENVOLVIMENTO 3

2.1 FUNÇÃO EXEMPLO 4

2.1.1 Resultados com o método BFGS 4

2.1.2 Resultados com o método DFP 5

2.1.3 Resultados com o método steepdesc 6

2.2 TAREFA 1 7

2.1.1 Resultados com o método BFGS 7

REFERÊNCIA BIBLIOGRÁFICA 8

ANEXO I 9

1 INTRODUÇÃO

O laboratório consiste na utilização do *solver* *fminunc*, presente no *Optimization Toolbox* do Matlab. Esta função calcula mínimos locais de funções com múltiplas variáveis, sem restrições e não-lineares, a partir de um ponto inicial informado. É possível utilizar os algoritmos *quasi-newton* ou *trust-region*, neste trabalho, o primeiro foi utilizado. Utilizando *quasi-newton*, é possível variar os métodos de escolha de direção de busca com os seguintes algoritmos: *bfgs, dfp* e *steepdesc*, foi feita uma comparação do comportamento de cada um deles para todas as funções objetivo.

Como a execução das tarefas era repetitiva, foi criada uma função auxiliar que recebe como parâmetros uma função objetivo e um ponto inicial e executa o *solver fminunc* utilizando *quasi-newton* e variando os métodos de busca. O código da função está disponível no Anexo I.

Para a execução das tarefas, foi utilizado o Matlab R2018b, versão 9.5.0 em ambiente *Windows*.

O código e os resultados obtidos nas execuções das tarefas também estão disponíveis no *Github* em https://github.com/alvarollmenezes/ifes-otimizacao-lab5.

2 DESENVOLVIMENTO

Entendidas as tarefas do laboratório, foi percebida a necessidade da execução de tarefas similares, variando-se as funções objetivo e o ponto inicial e utilizando-se os três algoritmos possíveis para os métodos de escolha de direção de busca. Sendo assim, foi criado o arquivo *executar.m* (Anexo I), contendo uma função que recebe como parâmetros a função a ser minimizada, o ponto inicial e retorna um *struct* com os resultados dos métodos *bfgs, dfp* e *steepdesc*.

Durante as execuções, foi utilizada a opção *Display* com o valor *iter*, desta forma a cada iteração o *solver* exibe uma saída com as informações dos valores calculados nesta iteração.

Nenhuma opção além de *Algorithm, hessupdate* e *Display* foi alterada, mantendo os padrões do Matlab no restante. Destaque para as opções *MaxIterations* que determina o número máximo de iterações permitido e tem como padrão 400 e *MaxFunctionsEvaluation* que determina o número máximo permitido de avaliações da função a ser minimizada, tendo como padrão 100 \* número de variáveis. Este último foi fator limitador em algumas execuções durante os testes. Todos os valores padrão utilizados pelo *solver* estão disponíveis na documentação do Matlab.

2.1 FUNÇÃO EXEMPLO

A primeira execução foi feita com a função f(x1,x2)=3(x1)2 + 2x1x2 + (x2)2, do texto explicativo, para isso foi criado o arquivo *fexemplo.m* (Anexo I), para ser utilizado na função *executar* com o ponto inicial em [1, 1], código executado:

>> [ex\_bfgs, ex\_dfp, ex\_steepdesc] = executar(@fexemplo, [1,1]);

2.1.1 Resultados com o método BFGS

Mínimo local encontrado em 8 iterações.

x1 = 2.5408e-07, x2 = -2.0293e-07.

fval = 1.3173e-13

Tabela 1 Iterações da função exemplo, utilizando BFGS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Iteration** | **Func-count** | **f(x)** | **Step-size** | **First-order optimality** |
| 0 | 3 | 6 |  | 8 |
| 1 | 6 | 0.25 | 0.125 | 1 |
| 2 | 9 | 0.0805763 | 1 | 0.389 |
| 3 | 12 | 0.0510334 | 1 | 0.276 |
| 4 | 15 | 0.00234029 | 1 | 0.158 |
| 5 | 18 | 0.000114716 | 1 | 0.0366 |
| 6 | 21 | 3.44E-07 | 1 | 0.00116 |
| 7 | 24 | 9.13E-10 | 1 | 4.45E-05 |
| 8 | 27 | 1.32E-13 | 1 | 1.16E-06 |

2.1.2 Resultados com o método DFP

Mínimo local encontrado em 8 iterações.

x1 = 2.5408e-07, x2 = -2.0293e-07.

fval = 1.3173e-13

Tabela 2 Iterações da função exemplo, utilizando DFP

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Iteration** | **Func-count** | **f(x)** | **Step-size** | **First-order optimality** |
| 0 | 3 | 6 |  | 8 |
| 1 | 6 | 0.25 | 0.125 | 1 |
| 2 | 12 | 0.0833333 | 0.166667 | 0.333 |
| 3 | 18 | 0.0277778 | 0.5 | 0.333 |
| 4 | 24 | 0.00925927 | 0.166667 | 0.111 |
| 5 | 30 | 0.00308643 | 0.499999 | 0.111 |
| 6 | 36 | 0.00102881 | 0.166667 | 0.037 |
| 7 | 42 | 0.000342938 | 0.499999 | 0.037 |
| 8 | 48 | 0.000114313 | 0.166667 | 0.0123 |
| 9 | 54 | 3.81E-05 | 0.499996 | 0.0123 |
| 10 | 60 | 1.27E-05 | 0.166668 | 0.00412 |
| 11 | 66 | 4.23E-06 | 0.499987 | 0.00412 |
| 12 | 72 | 1.41E-06 | 0.166669 | 0.00137 |
| 13 | 78 | 4.70E-07 | 0.499961 | 0.00137 |
| 14 | 84 | 1.57E-07 | 0.166674 | 0.000457 |
| 15 | 90 | 5.23E-08 | 0.499882 | 0.000457 |
| 16 | 96 | 1.74E-08 | 0.166689 | 0.000153 |
| 17 | 102 | 5.82E-09 | 0.499646 | 0.000153 |
| 18 | 108 | 1.94E-09 | 0.166735 | 5.09E-05 |
| 19 | 114 | 6.48E-10 | 0.49894 | 5.09E-05 |
| 20 | 120 | 2.16E-10 | 0.16687 | 1.71E-05 |
| 21 | 126 | 7.26E-11 | 0.496837 | 1.71E-05 |
| 22 | 132 | 2.43E-11 | 0.167278 | 5.77E-06 |

2.1.3 Resultados com o método steepdesc

Mínimo local encontrado em 22 iterações.

x1 = -2.8500e-06, x2 = 5.6879e-06.

fval = 2.4299e-11

Tabela 3 Iterações da função exemplo, utilizando steedesc

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Iteration** | **Func-count** | **f(x)** | **Step-size** | **First-order optimality** |
| 0 | 3 | 6 |  | 8 |
| 1 | 6 | 0.25 | 0.125 | 1 |
| 2 | 12 | 0.0833333 | 0.166667 | 0.333 |
| 3 | 18 | 0.0277778 | 0.5 | 0.333 |
| 4 | 24 | 0.00925927 | 0.166667 | 0.111 |
| 5 | 30 | 0.00308643 | 0.499999 | 0.111 |
| 6 | 36 | 0.00102881 | 0.166667 | 0.037 |
| 7 | 42 | 0.000342938 | 0.499999 | 0.037 |
| 8 | 48 | 0.000114313 | 0.166667 | 0.0123 |
| 9 | 54 | 3.81E-05 | 0.499996 | 0.0123 |
| 10 | 60 | 1.27E-05 | 0.166668 | 0.00412 |
| 11 | 66 | 4.23E-06 | 0.499987 | 0.00412 |
| 12 | 72 | 1.41E-06 | 0.166669 | 0.00137 |
| 13 | 78 | 4.70E-07 | 0.499961 | 0.00137 |
| 14 | 84 | 1.57E-07 | 0.166674 | 0.000457 |
| 15 | 90 | 5.23E-08 | 0.499882 | 0.000457 |
| 16 | 96 | 1.74E-08 | 0.166689 | 0.000153 |
| 17 | 102 | 5.82E-09 | 0.499646 | 0.000153 |
| 18 | 108 | 1.94E-09 | 0.166735 | 5.09E-05 |
| 19 | 114 | 6.48E-10 | 0.49894 | 5.09E-05 |
| 20 | 120 | 2.16E-10 | 0.16687 | 1.71E-05 |
| 21 | 126 | 7.26E-11 | 0.496837 | 1.71E-05 |
| 22 | 132 | 2.43E-11 | 0.167278 | 5.77E-06 |

2.2 TAREFA 1 – letra a

A execução desta tarefa foi feita com a função f(x1,x2) = 2(x1 - 2x2)2 + ½(x2 -(x1)2 )2, para isso foi criado o arquivo *ft1a.m* (Anexo I), para ser utilizado na função *executar* com o ponto inicial em [2, 6], código executado:

>> [ft1a\_bfgs, ft1a\_dfp, ft1a\_steepdesc] = executar(@ft1a, [2,6]);

2.2.1 Resultados com o método BFGS

Mínimo local encontrado em 15 iterações.

x1 = 0.5000, x2 = 0,2500.

fval = 2.1236e-12

Tabela 4 Iterações da função da Tarefa 1 - letra a, utilizando BFGS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Iteration** | **Func-count** | **f(x)** | **Step-size** | **First-order optimality** |
| 0 | 3 | 202 |  | 82 |
| 1 | 6 | 111.372 | 0.0121951 | 57.6 |
| 2 | 9 | 28.3976 | 1 | 28.2 |
| 3 | 12 | 7.61359 | 1 | 13.8 |
| 4 | 15 | 2.43697 | 1 | 7.04 |
| 5 | 18 | 0.791798 | 1 | 3.18 |
| 6 | 21 | 0.249344 | 1 | 1.34 |
| 7 | 24 | 0.0770373 | 1 | 0.574 |
| 8 | 27 | 0.0226955 | 1 | 0.25 |
| 9 | 30 | 0.0061752 | 1 | 0.107 |
| 10 | 33 | 0.00145044 | 1 | 0.0441 |
| 11 | 36 | 0.000259519 | 1 | 0.0165 |
| 12 | 39 | 2.77E-05 | 1 | 0.00505 |
| 13 | 42 | 1.12E-06 | 1 | 0.000998 |
| 14 | 45 | 7.67E-09 | 1 | 8.36E-05 |
| 15 | 48 | 2.12E-12 | 1 | 1.61E-06 |

2.2.2 Resultados com o método DFP

Mínimo local encontrado em 15 iterações.

x1 = 0.5000, x2 = 0,2500.

fval = 2.1236e-12

Tabela 5 Iterações da função da Tarefa 1 - letra a, utilizando DFP

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Iteration** | **Func-count** | **f(x)** | **Step-size** | **First-order optimality** |
| 0 | 3 | 202 |  | 82 |
| 1 | 6 | 111.372 | 0.0121951 | 57.6 |
| 2 | 9 | 28.3976 | 1 | 28.2 |
| 3 | 12 | 7.61359 | 1 | 13.8 |
| 4 | 15 | 2.43697 | 1 | 7.04 |
| 5 | 18 | 0.791798 | 1 | 3.18 |
| 6 | 21 | 0.249344 | 1 | 1.34 |
| 7 | 24 | 0.0770373 | 1 | 0.574 |
| 8 | 27 | 0.0226955 | 1 | 0.25 |
| 9 | 30 | 0.0061752 | 1 | 0.107 |
| 10 | 33 | 0.00145044 | 1 | 0.0441 |
| 11 | 36 | 0.000259519 | 1 | 0.0165 |
| 12 | 39 | 2.77E-05 | 1 | 0.00505 |
| 13 | 42 | 1.12E-06 | 1 | 0.000998 |
| 14 | 45 | 7.67E-09 | 1 | 8.36E-05 |
| 15 | 48 | 2.12E-12 | 1 | 1.61E-06 |

2.2.3 Resultados com o método steepdesc

Mínimo local não encontrado, foram executadas 31 iterações e o *solver* parou por extrapolar o número máximo de avaliações da função (200).

x1 = 0.5047, x2 = 0,2525.

fval = 2.6207e-06

Tabela 6 Iterações da função da Tarefa 1 - letra a, utilizando steepdesc

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Iteration** | **Func-count** | **f(x)** | **Step-size** | **First-order optimality** |
| 0 | 3 | 202 |  | 82 |
| 1 | 6 | 111.372 | 0.0121951 | 57.6 |
| 2 | 15 | 46.1924 | 0.0343196 | 41.2 |
| 3 | 27 | 30.7041 | 0.0805713 | 32.3 |
| 4 | 36 | 0.0277144 | 0.0485361 | 0.744 |
| 5 | 45 | 0.00848396 | 0.0450154 | 0.079 |
| 6 | 48 | 0.00346415 | 1 | 0.2 |
| 7 | 57 | 0.00197988 | 0.0462287 | 0.0304 |
| 8 | 60 | 0.0012058 | 1 | 0.101 |
| 9 | 69 | 0.000810133 | 0.0467636 | 0.0175 |
| 10 | 72 | 0.00055928 | 1 | 0.0653 |
| 11 | 81 | 0.000394739 | 0.0469457 | 0.0115 |
| 12 | 84 | 0.000283495 | 1 | 0.0431 |
| 13 | 93 | 0.000210698 | 0.0471938 | 0.008 |
| 14 | 96 | 0.000158218 | 1 | 0.0317 |
| 15 | 105 | 0.000118943 | 0.0472219 | 0.00584 |
| 16 | 108 | 9.01E-05 | 1 | 0.0227 |
| 17 | 117 | 6.97E-05 | 0.0473911 | 0.00435 |
| 18 | 120 | 5.42E-05 | 1 | 0.0177 |
| 19 | 129 | 4.19E-05 | 0.0473621 | 0.00332 |
| 20 | 132 | 3.26E-05 | 1 | 0.0131 |
| 21 | 141 | 2.57E-05 | 0.0474981 | 0.00256 |
| 22 | 144 | 2.04E-05 | 1 | 0.0105 |
| 23 | 153 | 1.60E-05 | 0.0474417 | 0.002 |
| 24 | 156 | 1.26E-05 | 1 | 0.00799 |
| 25 | 165 | 1.01E-05 | 0.0475608 | 0.00157 |
| 26 | 168 | 8.06E-06 | 1 | 0.00649 |
| 27 | 177 | 6.39E-06 | 0.0474896 | 0.00124 |
| 28 | 180 | 5.08E-06 | 1 | 0.005 |
| 29 | 189 | 4.08E-06 | 0.0475992 | 0.000985 |
| 30 | 192 | 3.29E-06 | 1 | 0.0041 |
| 31 | 201 | 2.62E-06 | 0.0475194 | 0.000788 |

2.3 TAREFA 1 – letra b

A execução desta tarefa foi feita com a função f(x1,x2) = ½( (x1-1)2 + (x2-(x1)2 )2 ), para isso foi criado o arquivo *ft1b.m* (Anexo I), para ser utilizado na função *executar* com o ponto inicial em [-1, 0], código executado:

>> [ft1b\_bfgs, ft1b\_dfp, ft1b\_steepdesc] = executar(@ft1b, [-1,0]);

2.3.1 Resultados com o método BFGS

Mínimo local encontrado em 10 iterações.

x1 = 1.0000, x2 = 1,0000.

fval = 5.6093e-13

Tabela 7 Iterações da função da Tarefa 1 - letra b, utilizando BFGS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Iteration** | **Func-count** | **f(x)** | **Step-size** | **First-order optimality** |
| 0 | 3 | 2.5 |  | 4 |
| 1 | 6 | 0.53125 | 0.25 | 1 |
| 2 | 9 | 0.205528 | 1 | 0.648 |
| 3 | 15 | 0.111588 | 0.430076 | 0.282 |
| 4 | 18 | 0.0918444 | 1 | 0.249 |
| 5 | 21 | 0.00954911 | 1 | 0.0541 |
| 6 | 24 | 0.00129009 | 1 | 0.0424 |
| 7 | 27 | 0.000350366 | 1 | 0.0586 |
| 8 | 30 | 8.07E-06 | 1 | 0.00702 |
| 9 | 33 | 1.26E-08 | 1 | 0.000183 |
| 10 | 36 | 5.61E-13 | 1 | 2.21E-06 |

2.3.2 Resultados com o método DFP

Mínimo local encontrado em 10 iterações.

x1 = 1.0000, x2 = 1,0000.

fval = 5.6093e-13

Tabela 8 Iterações da função da Tarefa 1 - letra b, utilizando DFP

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Iteration** | **Func-count** | **f(x)** | **Step-size** | **First-order optimality** |
| 0 | 3 | 2.5 |  | 4 |
| 1 | 6 | 0.53125 | 0.25 | 1 |
| 2 | 9 | 0.205528 | 1 | 0.648 |
| 3 | 15 | 0.111588 | 0.430076 | 0.282 |
| 4 | 18 | 0.0918444 | 1 | 0.249 |
| 5 | 21 | 0.00954911 | 1 | 0.0541 |
| 6 | 24 | 0.00129009 | 1 | 0.0424 |
| 7 | 27 | 0.000350366 | 1 | 0.0586 |
| 8 | 30 | 8.07E-06 | 1 | 0.00702 |
| 9 | 33 | 1.26E-08 | 1 | 0.000183 |
| 10 | 36 | 5.61E-13 | 1 | 2.21E-06 |

2.3.3 Resultados com o método steepdesc

Mínimo local não encontrado, foram executadas 34 iterações e o *solver* parou por extrapolar o número máximo de avaliações da função (200).

x1 = 0.9780, x2 = 0,9430.

fval = 3.3330e-04

Tabela 9 Iterações da função da Tarefa 1 - letra b, utilizando steepdesc

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Iteration** | **Func-count** | **f(x)** | **Step-size** | **First-order optimality** |
| 0 | 3 | 2.5 |  | 4 |
| 1 | 6 | 0.53125 | 0.25 | 1 |
| 2 | 12 | 0.114011 | 0.593811 | 0.251 |
| 3 | 15 | 0.0542461 | 1 | 0.14 |
| 4 | 18 | 0.0310201 | 1 | 0.153 |
| 5 | 24 | 0.0251242 | 0.457189 | 0.145 |
| 6 | 30 | 0.0204597 | 0.389164 | 0.12 |
| 7 | 36 | 0.0169137 | 0.443633 | 0.117 |
| 8 | 42 | 0.0140344 | 0.371364 | 0.0973 |
| 9 | 48 | 0.0117675 | 0.433582 | 0.0966 |
| 10 | 54 | 0.00989326 | 0.357906 | 0.0802 |
| 11 | 60 | 0.00838137 | 0.425864 | 0.0807 |
| 12 | 66 | 0.00711492 | 0.347406 | 0.0669 |
| 13 | 72 | 0.00607504 | 0.419782 | 0.0681 |
| 14 | 78 | 0.0051953 | 0.33902 | 0.0565 |
| 15 | 84 | 0.00446319 | 0.414894 | 0.058 |
| 16 | 90 | 0.00383904 | 0.332202 | 0.0481 |
| 17 | 96 | 0.00331418 | 0.410902 | 0.0497 |
| 18 | 102 | 0.00286396 | 0.32658 | 0.0411 |
| 19 | 108 | 0.00248222 | 0.407602 | 0.0428 |
| 20 | 114 | 0.00215313 | 0.321893 | 0.0354 |
| 21 | 120 | 0.00187222 | 0.404846 | 0.0371 |
| 22 | 126 | 0.00162908 | 0.317948 | 0.0306 |
| 23 | 132 | 0.00142038 | 0.402524 | 0.0322 |
| 24 | 138 | 0.00123913 | 0.314604 | 0.0265 |
| 25 | 144 | 0.00108285 | 0.400555 | 0.028 |
| 26 | 150 | 0.000946733 | 0.311752 | 0.0231 |
| 27 | 156 | 0.00082892 | 0.398874 | 0.0245 |
| 28 | 162 | 0.000726067 | 0.309305 | 0.0201 |
| 29 | 168 | 0.000636757 | 0.397432 | 0.0214 |
| 30 | 174 | 0.000558628 | 0.307198 | 0.0176 |
| 31 | 180 | 0.0004906 | 0.396189 | 0.0187 |
| 32 | 186 | 0.000430987 | 0.305377 | 0.0154 |
| 33 | 192 | 0.000378959 | 0.395115 | 0.0164 |
| 34 | 198 | 0.000333299 | 0.303797 | 0.0135 |

REFERÊNCIA BIBLIOGRÁFICA

BELLOC, M.; VELASTIN, S.; FERNANDEZ, R.; JARA, M. Detection of People Boarding/Alighting a Metropolitan Train using Computer Vision. 9th International Conference on Pattern Recognition Systems (ICPRS 2018), pp. 5-11. 2018.

BRASIL. Ministério das Cidades. Política Nacional de Mobilidade Urbana Sustentável: Princípios e Diretrizes. Brasília. 2004.

CHATO, P.; CHIPANTASI, D.; VELASCO, N.; REA, S.; HALLO, V.; CONSTANTE, P. Image processing and artificial neural network for counting people inside public transport. IEEE Third Ecuador Technical Chapters Meeting (ETCM), pp. 1-5. 2018.

COCCHIA, A. Smart and digital city: a systematic literature review. Smart City: How to Create Public and Economic Value with High Technology in Urban Space, 13-43. 2014.

GUERRA, A. L.; BARBOSA, H. M.; OLIVEIRA, L. K. Estimativa de Matriz Origem/Destino Utilizando Dados do Sistema de Bilhetagem Eletrônica: Proposta Metodológica. Transportes, [s.l.], v 22, n. 3, p. 26-38, 2014.

PRADO, A.L.; DA COSTA, E.M.; FURLANI, T.Z.; YIGITCANLAR, T. Smartness that matters: Towards a comprehensive and human-centred characterisation of smart cities. Journal of Open Innovation Technology, Market and Complexity. 2016.

MOREIRA-MATIAS L.; MENDES-MOREIRA J.; DE SOUSA, J.; GAMA, J. Improving Mass Transit Operations by Using AVL-Based Systems: A Survey. IEEE Transactions on Intelligent Transportation Svstems, vol. 16, no. 4, pp. 1636-1653, 2015.

MOREIRA, Mauricio Renato Pina; SCHREINER, Sideney. Pesquisas de Origem e Destino na Região Metropolitana do Recife. In: XXXI Congresso Nacional de Pesquisa em Transportes da ANPET. Recife, 2017. p. 2039-2046.

REDMON, J.; DIVVALA, S.; GIRSHICK, R.; FARHADI, A. You Only Look Once: Unified, Real-Time Object Detection. arXiv. 2016.

REDMON, J.; FARHADI, A. YOLOv3: An Incremental Improvement. arXiv. 2018.

SARAIVA, M. A Cidade e o Tráfego Uma Abordagem Estratégica. Recife. Ed. Universitária da UFPE. 2000.

ANEXO I

Arquivo executar.m:

function [bfgs, dfp, steep] = executar(func, x0)

options\_bfgs = optimoptions(@fminunc, 'Algorithm', 'quasi-newton',

'hessupdate', 'bfgs', 'Display', 'iter');

options\_dfp = optimoptions(@fminunc, 'Algorithm', 'quasi-newton',

'hessupdate', 'bfgs', 'Display', 'iter');

options\_steep = optimoptions(@fminunc, 'Algorithm', 'quasi-newton',

'hessupdate', 'steepdesc', 'Display', 'iter');

disp('Executando ''fminunc'' com o método ''bfgs''');

[bfgs.x, bfgs.fval, bfgs.exitflag, bfgs.output] = ...

fminunc(func, x0, options\_bfgs);

bfgs

bfgs\_output = bfgs.output

disp('Executando ''fminunc'' com o método ''dfp''');

[dfp.x, dfp.fval, dfp.exitflag, dfp.output] = ...

fminunc(func, x0, options\_dfp);

dfp

dfp\_output = dfp.output

disp('Executando ''fminunc'' com o método ''steepdesc''');

[steep.x, steep.fval, steep.exitflag, steep.output] = ...

fminunc(func, x0, options\_steep);

steep

steep\_output = steep.output

end

Arquivo fexemplo.m:

function f = fexemplo(x)

f = 3\*x(1)^2 + 2\*x(1)\*x(2) + x(2)^2;

end

Arquivo ft1a.m:

function f = ft1a(x)

f = 2 \* (x(1) - 2\*x(2))^2 + 1/2 \* (x(2) - x(1)^2)^2;

end

Arquivo ft1b.m:

function f = ft1b(x)

f = 1/2 \* ((x(1) - 1)^2 + (x(2) - x(1)^2)^2);

end

Arquivo ft1c.m:

function f = ft1c(x)

f = -24\*x(1)\*x(2) + x(1)^2\*x(2) + x(1)\*x(2)^2;

end

Arquivo ft2i.m:

function f = ft2i(x)

f = x(1)^2 + x(2)^2 - 1;

end

Arquivo ft2ii.m:

function f = ft2ii(x)

f = x(1)^4 + x(1)^2 + x(2)^2;

end

Arquivo ft2iii.m:

function f = ft2iii(x)

f = (-13 + x(1) + 5\*x(2)^2 - x(2)^3 - 2\*x(2))^2 + ...

(-29 + x(1) + x(2)^3 + x(2)^2 - 14\*x(2))^2;

end

Arquivo ft2iv.m:

function f = ft2iv(x)

f = 100 \* (x(2) - x(1)^2)^2 + (1 - x(1))^2;

end