Technische Universität Chemnitz
Prof. Dr. R. Herzog, J. Blechschmidt, A. Schäfer

Chemnitz, 24. Oktober 2014
Abgabe am 7. November 2014

Grundlagen der Optimierung

Übung 2

Aufgabe 5: Einführung in `Matlab`

Interpretieren Sie die Bildschirm-Ausgaben nach Eingabe der folgenden `Matlab`-Anweisungen. Achten Sie auf unterschiedliche Eingabemöglichkeiten (Bedeutung von `,` `;` `:`) und deren Auswirkungen aufs Ergebnis.

Hinweis: Mit den Cursortasten $\boxed{\uparrow}$, $\boxed{\downarrow}$ kann man die vorherige Eingabezeilen „zurückholen“ und ändern)

(a) **Eingabe, Verwendung von Variablen, Matrizen und Vektoren**

```
≫ pwd
≫ cd matlab
≫ 3*4
≫ pi/2
≫ A = [1 2 3 4 5 6 7 8 9]
≫ A = [1 2 3; 4 5 6; 7 8 9]
≫ M = magic(6)
≫ B = [4, 5, 6.1], x=[ -1.3  sqrt(3)  (1+2+3)^3 ]
≫ E = ones(3), F = zeros(3)
≫ e = ones(1,3), f = zeros(3,1)
≫ G = 3*diag(e) - 2*E
≫ E * G, E .* G, p1 = e*e', p2 = e'*e
≫ x(8) = -x(2); y=x';
```

(b) **Ausgabe von Ergebnissen in unterschiedlichem Format**

Standardausgabeformat mit Variablennamen

```
≫ display(3*4)
```

nur Werte ohne Variablennamen

```
≫ disp(3*4)
≫ disp('     A         B'), disp(rand(4,2))
```

mit Textformatierung

```
≫ fprintf('Die Zahl pi lautet: %20.16f...\n',pi)
≫ pause; disp(1); pause(5); disp(2); pause(0)
```

(c) **Verschiedene Zahlendarstellungen**

```
≫ x, y
```

≫ `format long`
≫ `x`
≫ `format long e`
≫ `x`
≫ `format +`
≫ `x`
≫ `format short`
≫ `omega = sqrt(-3)`
≫ `z1 = 10^20`
≫ `z2 = 1e20`
≫ `s=1/0, t=1/s, u=t*s`

(d) **Verwendung des Doppelpunktes (Zähl-Intervalle, Indexbereiche)**

≫ `n=10`
≫ `1:n`
≫ `1:2:n`
≫ `2:2:n`
≫ `linspace(0,10,6), cos(linspace(0,pi,5))`
≫ `y = x(3:5)`
≫ `y(4:6)=x(1:3)`
≫ `r = [10 11 12]`
≫ `q = [13:16]', p = [13;14;15;16]`
≫ `B = [ A; r ]`
≫ `C = [ B p ]`
≫ `C = C(:,4:-1:1)`
≫ `A = C(1:2,:)`
≫ `size(B)`
≫ `A = C(:,1:3)`
≫ `[n,m] = size(A)`
≫ `v = A(:), w=reshape(A,1,m*n)`
≫ `x = pi*(0:1/2:2)`
oder
≫ `x=linspace(0,2*pi,5)`
≫ `s = sin(x), c = cos(x)`

(e) **Eigene Funktionen, Schleifen, Verzweigungen**

ein sogenanntes „M-File“ anlegen (auch mit beliebigen Editor)
≫ `edit potmat.m`

Inhalt der Datei `potmat.m`**:**

```
function A=potmat(x,n)
% potmat(x,n) - erzeugt eine m-kreuz-n Matrix deren
%               Spalten die elementweisen Potenzen
%               des Vektors x sind
%               A= [1 x x.^2 ... x.^(n-1)]
%
```

```
if ( (n < 1) || (size(x,2) ~= 1) )
  error('n >= 1 ist erforderlich, x muss Spaltenvektor sein');
else
  m = length(x);
  A = zeros(m,n);
  for i=1:n
    A(:,i) = x.^(i-1);
  end

  return;
end
```

Verwenden der Funktion `potmat`**:**
selbst eingebauten Hilfetext anzeigen

$\gg$ `help potmat`
$\gg$ `C=potmat([0;1;2;3],3)`

(f) **Rechnen mit Matrizen**

$\gg$ `v=logspace(-1, 1, 3)'`
$\gg$ `D=diag(v)` — Diagonalmatrix
$\gg$ `A=C+D;` — ! Dimensionen passen nicht
$\gg$ `B=C(1:3,:)` — ! Dimensionen passen nicht
$\gg$ `A=B+D,`
$\gg$ `E=eig(A)` — Eigenwerte von C
$\gg$ `[V,D]=eig(A)` — Eigenvektoren $V_1, \ldots, V_n$ und Diagonalmatrix der EW
$\gg$ `diag(D)`
$\gg$ `diag(E)`
$\gg$ `W=A*V-V*D` — sollte bei exakter Rechnung Null sein
$\gg$ `max(abs(W))`
$\gg$ `max(max(abs(W)))`
$\gg$ `c=rand(3,1)` — Zufallsvektor
$\gg$ `b=A*c`

Löse lineares Gleichungssystem $Ax = b$

$\gg$ `x=A^(-1)*b`

oder

$\gg$ `x=inv(A)*b`

aber besser

$\gg$ `x=A\b`

oder dasselbe

$\gg$ `x=(b'/A')'`
$\gg$ `x-c` — sollte bei exakter Rechnung Null sein

(g) **Grafische Darstellungen**

$\gg$ `x=linspace(0,2*pi); c=cos(x); s=sin(x);`
$\gg$ `plot(c)`

≫ `plot(x,c)`	
≫ `subplot(2,1,1),plot(x,c),title('cos')`	
≫ `subplot(2,1,2),plot(x,s),title('sin')`	
≫ `subplot(1,1,1),plot(c,s)`	
≫ `axis equal`	
≫ `figure(2)`	
≫ `plot(c), hold on, plot(s), hold off`	
≫ `plot(x,c,x,s)`	
≫ `plot(x,c,'o',x,s,':')`	
≫ `Y = [ c; s ]; plot(x,Y)`	
≫ `title('Funktionen')`	
≫ `xlabel('x-Achse')`	
≫ `ylabel('f(x)')`	
≫ `grid`	
≫ `x=-8:0.5:8; y=x';`	
≫ `X=ones(size(y))*x; Y=y*ones(size(x));`	
≫ `R=sqrt(X.^2+Y.^2)+eps;`	
≫ `Z=sin(R)./R;`	
≫ `mesh(Z);`	
≫ `view(45,45)`	3D-Ansicht (Winkel in Kugelkoordinaten)
≫ `view(0,90)`	
≫ `view(45,90)`	
≫ `view(0,0)`	
≫ `view(3)`	Standardwerte für 3D-plot: $(-37.5, 30)$

(h) **Hilfsfunktionen**

≫ `diary`	Ein-/Ausschalten der Protokollierung (in eine Datei)
≫ `who`	Liste aller Variablen
≫ `whos`	Variablenliste mit Speicherbedarf
≫ `save temp`	alle Variablen retten als `'temp.mat'`
≫ `clear; whos`	alle Variablen löschen
≫ `load temp A; whos`	nur Matrix A wieder einlesen
≫ `load temp`	alle geretteten Variablen wiederherstellen
≫ `who`	
≫ `save tmpmat A`	nur die Matrix A speichern
≫ `save a.dat A -ascii`	dasselbe, aber lesbar
≫ `dir`	
≫ `clear A; who`	
≫ `load a.dat; who`	Matrix A heißt jetzt a

Hilfe allgemein und zu einzelnen Kommandos:

≫ `help`
≫ `help [`
≫ `help punct`

```
≫ help inv
≫ doc det
≫ help diag
≫ lookfor root
```

(i) **Mitgelieferte Demos**

```
≫ intro
≫ demo
```

Aufgabe 6: Gradient und Hessematrix der Himmelblau-Funktion

Bestimmen Sie den Gradienten und die Hessematrix der Himmelblau-Funktion

$$f(x_1, x_2) = \left(x_1^2 + x_2 - 11\right)^2 + \left(x_1 + x_2^2 - 7\right)^2$$

und zeichnen Sie diese mit `Matlab` in $[-6, 6] \times [-6, 6]$.

Aufgabe 7: Projektionsaufgabe

Die (orthogonale) Projektion eines Punktes $p \in \mathbb{R}^m$ auf eine abgeschlossene konvexe Menge $C \subset \mathbb{R}^m$ mit $C \neq \emptyset$ ist der Punkt $\hat{x} \in C$, welcher das Optimierungsproblem

$$\min \|x - p\| \quad \text{über } x \in C$$

löst ($\|\cdot\|$ bezeichnet die euklidische Norm).

(a) Zeige, dass die Projektion wohldefiniert ist (und somit eine Funktion $\mathrm{proj}_C(\cdot) : \mathbb{R}^m \to C$ definiert).

(b) Bestimme eine explizite Formel für die Projektion auf einen linearen Unterraum, der durch $\{A^T y : y \in \mathbb{R}^n\}$ mit $A \in \mathbb{R}^{n \times m}$ ($n \leq m$, voller Zeilenrang) gegeben ist.
Hinweis: Betrachte das Optimierungsproblem $\min \frac{1}{2}\|A^T y - p\|^2$.
Warum darf man hier das Quadrat der Zielfunktion verwenden?

Hinweis: Eine Menge $C \subset \mathbb{R}^n$ heißt **konvex**, wenn mit $x, y \in C$ und $\lambda \in [0, 1]$ auch $\lambda x + (1 - \lambda)y \in C$ liegt.

Aufgabe 8: Standortoptimierung

Der Standort einer Rettungswache mit Hubschrauber soll geplant werden. Sie soll m Ortschaften versorgen. Maß für die Güte eines Standorts ist die gewichtete Summe der Abstände des Standorts zu den Ortschaften (je kleiner, desto besser). Dabei sind die Gewichte der Ortschaften proportional zu ihrer jeweiligen Bevölkerungszahl. Die Luftfahrtbehörde erlaubt einen Hubschrauberlandeplatz nur im Kreissektor $K := \{x \in \mathbb{R}^2 : x \geq 0,\ \|x\|_2 \leq 1\}$. Stelle ein passendes mathematisches Modell auf.

Hausaufgabe 5: Bestimmung von Extrempunkten in Abhängigkeit von c

Bestimmen Sie die Extrempunkte von

$$f : \mathbb{R}^2 \to \mathbb{R}, (x, y) \mapsto x^4 + y^4 - 4c^2xy$$

mit $c \in \mathbb{R}$. Erstellen Sie für ausgewählte Parameter c einen Plot der Zielfunktion (z. B. in Matlab mit Höhenlinien).

Hausaufgabe 6: Beispiel: Notwendige Bedingungen zweiter Ordnung sind nicht hinreichend

Besitzt die Funktion $f(x) = (x_1 - x_2^2)\,(2x_1 - x_2^2)$ im Punkt $x^* = (0, 0)$ ein lokales Minimum? Begründe deine Antwort.

Hausaufgabe 7: Einschachtelungsargument

Es sei $\{f_k\}_{k\geq 0} \subset \mathbb{R}$ eine monoton fallende Folge mit einer gegen f^* konvergenten Teilfolge, d.h. $f_{k_m} \to f^*$. Zeige, dass sogar die ganze Folge f_k gegen f^* konvergiert.

Hausaufgabe 8: Rucksackproblem (Knapsack Problem)

Fritzchen überlegt, welche der Bücher $1, \dots, n$ er aus der Bibliothek mit nach Hause nimmt. Jedes Buch $i \in \{1, \dots, n\}$ hat einen Nutzen $p_i \geq 0$ für Fritzchen und ein Gewicht $w_i \geq 0$. Fritzchen kann ein maximales Gewicht W tragen. Welche Bücher sollte er mitnehmen, damit der Gesamtnutzen (=Summe der Nutzen der mitgenommenen Bücher) maximal ist? Formuliere eine geeignete Optimierungsaufgabe.

Hausaufgabe 9: Das Rucksackproblem mit AMPL

Entwickeln Sie ein AMPL-Modelfile `knapsack.mod` und ein AMPL-Datafile `knapsack.dat` für das Knapsackproblem aus Hausaufgabe 8 mit den Daten $n = 10$, $p = (5\ \ 4\ \ 7\ \ 1\ \ 2\ \ 3\ \ 4\ \ 4\ \ 6\ \ 8)^\top$, $w = (10\ \ 9\ \ 8\ \ 7\ \ 6\ \ 5\ \ 4\ \ 3\ \ 2\ \ 1)^\top$ und $W = 35$. Nutze `knapsack.cmd` von der Lehrveranstaltungsseite, um die Instanz über NEOS mit MINTO und MINOS zu lösen. Interpretiere die Ergebnisse und schicke die MINTO/MINOS-Lösungen, `knapsack.mod` und `knapsack.dat` an `jan.blechschmidt@mathematik.tu-chemnitz.de` (Betreff: HA Grundlagen der Optimierung Übung 2)!