8 Softwarebibliotheken in Java

8.1 Packages

Erste Anweisung: package pfad.zum.paket;

- dient zur Organisation von zusammengehörigen Unterprogrammen, Objekten und Daten in `.jar`–Archiven.
- Objekte innerhalb eines Package haben Zugriffsprivilegien gegeneinander.
- Namensschema für Nicht–System–Pakete: land.organ. titel , für HUMath, wenn es denn frei veröffentlicht würde, also korrekter de.humath.algebra, de.humath.numerik.
- ch.aplu. turtle entspricht dem Schema.
- Systempakete sind java.lang, immer implizit eingebunden, java.io für Dateien und Verzeichnisse, java. text für Stringformatierung, java.math für BigDecimal und BigInteger.

8.2 HUMath

Intern zum Lernen, nicht zur Weitergabe. Vollständig unter `/usr/math/java/lib/HUMath`. Für den Numerik-Teil ist die Jama-Matrixbibliothek erforderlich.

HUMath.Algebra enthält die Klassen

B	diverse Hilfsfunktionen
C	komplexe Zahlen
DX, CX, QX	Polynome mit reellen, komplexen und rationalen Koeffizienten
DM, CM, QM	Matrizen mit reellen, komplexen und rationalen Einträgen
DXM, CXM, QXM	Matrizen mit polynomialen Einträgen
Z	ganze Zahlen und Zahlentheorie
P	Restklassenarithmetik
...	

8.3 Gauß-Algorithmus

$Ax = b \iff a_{i1}x_1 + \cdots + a_{in}x_n = b_i,\ i = 1, \ldots, m$

Skalieren von Zeilen und Addieren einer Zeile zu einer anderen sind äquivalente Umformungen.

```
for(int i=1; i<=m; i++){
      \\ a[i][1..i-1]=0 nach Rechnung bisher
  double x=a[i][i]; a[i][i]=1;
      \\ bringe Zeile i auf Normalform
      for(int j=i+1; j<=n; j++) a[i][j]/=x;
      for(int k=i+1;k<=m;k++){
        double y=a[k][i]; a[k][i]=0;
        \\ reduziere Zeile k
        for(int j=i+1;j<=n;j++) a[k][j]-=y*a[i][j];
      }
}
```

Dies setzt voraus, dass die Diagonalelemente `a[i][i]`, die im Laufe der Rechnung benutzt werden, stets deutlich von Null verschieden sind. Ist `x=a[i][i]` sehr klein relativ zu den anderen Matrixeinträgen, so entstehen bei Division durch `x` sehr große Zwischenergebnisse. Dies lässt sich nicht immer verhindern, aber auf die letzten Zeilen hinauszögern. Dazu wird in der aktuellen `i`-ten Spalte (von Zeile `i` bis `m`) das betragsgrößte Element gesucht und dessen Zeile an die Position `i` getauscht. Generell kann man auch in der nächsten `i+1`-ten Spalte oder der gesamten Teilmatrix ab Zeile und Spalte `i` nach dem größten Element suchen und dann auch die entsprechende Spalte an Position `i` tauschen. Dazu ist eine Buchführung über Zeilen- und Spaltenpermutationen erforderlich, um am Ende das Ergebnis richtig interpretieren zu können.

8.4 Newton–Verfahren

$f : [a,b] \to \mathbb{R}$ sei wenigstens einmal differenzierbar.

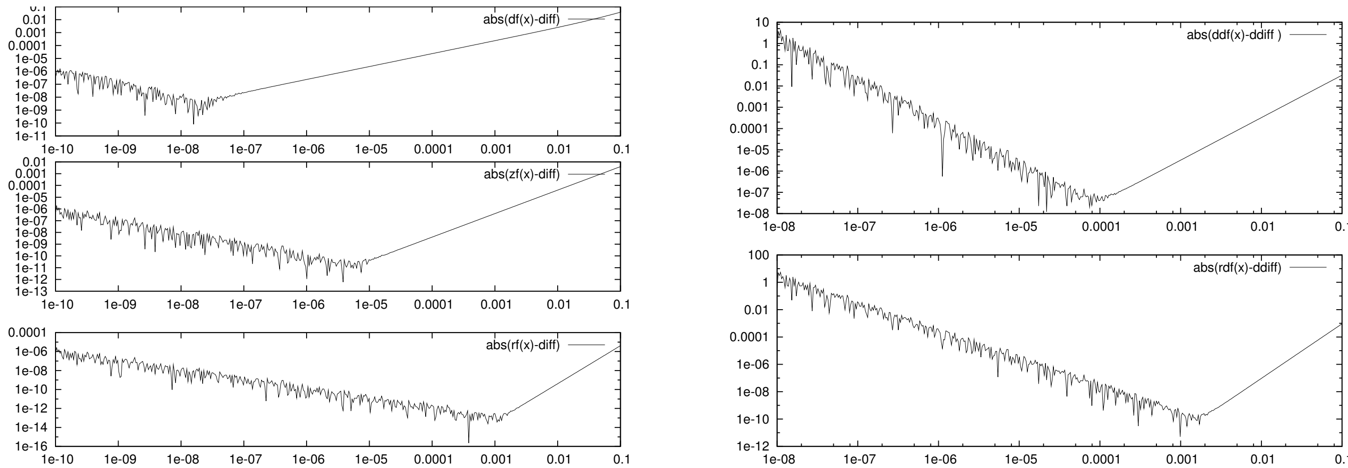
Für x_0 nahe einer Nullstelle x^* liegen f und die Tangente t in x_0,

$$t(x) = f(x_0) + f'(x_0)(x - x_0),$$

sehr dicht beieinander. Löse $t(x_1) = 0$

$$x_1 = x_0 - \frac{f(x)}{f'(x)}$$

Wir benötigen also die Ableitung von f. Drei Möglichkeiten: Ableitungsausdruck von Hand bzw. symbolisch ausrechnen, mit Differenzenquotienten approximieren, mit algorithmischem Differenzieren parallel zur Funktion auswerten.



Links: einfacher und zentraler Differenzenquotient sowie die Extrapolation.
Rechts: Differenzenquotient zweiter Ordnung und Extrapolation.

Abbildung 1: Ableitungen mittels Differenzenquotienten

8.5 Numerisches Differenzieren

Benutze *Differenzenquotienten* als Näherung der Ableitung. Die einfache Formel des rechtsseitigen Quotienten

$$\frac{f(x+h)-f(x)}{h} = f'(x) + \frac{1}{2}f''(x)h + O(h^2)$$

ergibt einen Fehler proportional h.

Die Auswertung der Funktion f ergibt einen Fehler proportional zur Maschinengenauigkeit μ. Die Größe kann als durch $c\mu$ beschränkt angenommen werden, wobei c etwa die Rechentiefe (Tiefe des Berechnungsgraphen) des Auswertungsalgorithmus ist.

Der Auswertungsfehler des Quotienten ist also beschränkt durch $\frac{2c\mu}{h}$. Der Gesamtfehler bzw. dessen Schranke

$$\varepsilon(h) = \frac{1}{2}|f''(x)|h + \frac{2c\mu}{h}$$

wird minimal, wenn beide Summanden gleichgroß sind (AGM-UG), also für $h = 2\sqrt{\frac{c\mu}{|f''(x)|}}$. Beide Fehler sind in etwa gleichgroß und daher in der Summe minimal für $h \sim \sqrt{\mu}$ mit einem Wert $\varepsilon \sim \sqrt{\mu}$.

Eine bessere Genauigkeit liefert der *zentrale Differenzenquotient*:

$$\frac{f(x+h)-f(x-h)}{2h} = f'(x) + \frac{1}{6}f'''(x)h^2 + O(h^4)$$

in diesem heben sich die geraden Terme der Taylorentwicklung weg. Eine Schranke für die Summe aus Approximations- und Rundungsfehler ist also

$$\varepsilon(h) = \frac{1}{6}|f'''(x)h^2| + \frac{2c\mu}{h}$$

Dieser Ausdruck wird minimal für $h = \sqrt[3]{\frac{6c\mu}{|f'''(x)h^2|}}$, also $h \sim \sqrt[3]{\mu}$ mit Wert $\varepsilon = 3\sqrt[3]{\frac{1}{6}|f'''(x)h^2|c^2\mu^2}$ also $\varepsilon \sim \sqrt[3]{\mu}^2$.

Eine ähnliche Rechnung für den *zentralen Differenzenquotienten zweiter Ordnung* liefert

$$\frac{f(x+h)-2f(x)+f(x-h)}{h^2} = f''(x) + \frac{1}{12}f^{(4)}(x)h^2 + O(h^4)$$

Durch Kombination der Differenzenquotienten mit Schrittweite h und $2h$ kann der nächsthöhere Taylorkoeffizient eliminiert werden. Das allgemeine Verfahren dazu wird *Richardson–Extrapolation* genannt.

In Abbildung 1 wird jeweils doppelt logarithmisch der Abstand des Differenzenquotienten zur tatsächlichen Ableitung dargestellt. Trotz von oben nach unten zunehmender Schrittweite des Optimums wird der Fehler im Minimum kleiner. Es ist in jedem Graphen zu erkennen, dass die rechte, ansteigende Seite vom analytischen und die linke, zerfasert fallende Seite vom Gleitkommafehler verursacht wird.