

**week5: Beispiele Rechnen mit Vektoren und Matrizen:  
Effizientes Matrix-Exponential**

Wir wollen die folgende Identität numerisch in R überprüfen:

$$\exp\left\{\varepsilon t \begin{pmatrix} 0 & +1 \\ -1 & 0 \end{pmatrix}\right\} = \begin{pmatrix} \cos \varepsilon t & \sin \varepsilon t \\ -\sin \varepsilon t & \cos \varepsilon t \end{pmatrix} \quad (1)$$

etwa für die Parameterwerte  $\varepsilon = 2$  und  $t \in [0, T] = [0, 15]$ . Dazu wollen wir jetzt eine effiziente Funktion zur Berechnung des Matrix-Exponentials codieren. Um den Algorithmus zu motivieren, gehen wir wieder in mehreren Schritten vor:

- a) Es sei  $i = \sqrt{-1}$  die komplexe imaginäre Einheit. Lösen Sie numerisch die DGL

$$\dot{z}_t = i\varepsilon z_t \quad (2)$$

auf dem Intervall  $t \in [0, T] = [0, 15]$  für  $\varepsilon = 2$  und  $z_0 = 1$ . Benutzen Sie dazu die Iteration

$$z_{t+dt} = z_t + i\varepsilon z_t dt = (1 + i\varepsilon dt) z_t \quad (3)$$

und plotten Sie die  $z_t$  in der komplexen Ebene. Schauen Sie sich die Bilder für verschiedene  $dt$ 's an und fügen Sie ebenfalls die exakte, theoretische Lösung

$$z_{\text{exact}}(t) = e^{i\varepsilon t} z_0 \quad (4)$$

dem Plot hinzu. Das R kann mit komplexen Zahlen rechnen, das  $i = \sqrt{-1}$  bekommen Sie durch die Zeichenkette `1i` und die vier Grundrechenarten sowie elementare Funktionen wie `exp`, `sin`, `cos`, `log` können damit rechnen.

- b) Berechnen Sie numerisch die Partialsummen

$$s_n(t) := \sum_{k=0}^n \frac{(i\varepsilon t)^k}{k!} \quad (5)$$

für  $n \in \{25, 50, 75, 100\}$  und  $t \in [0, T] = [0, 15]$  und plotten Sie die  $s_n(t)$  in der komplexen Ebene.

- c) Berechnen Sie den Ausdruck

$$e_n(t) := \left(1 + \frac{i\varepsilon t}{n}\right)^n \quad (6)$$

für  $n \in \{100, 1000, 10000, 100000\}$  und  $t \in [0, T] = [0, 15]$  und plotten Sie die  $e_n(t)$  in der komplexen Ebene.

d) Berechnen Sie die Matrix

$$E_n(t) := \left\{ Id + \frac{\varepsilon t}{n} \begin{pmatrix} 0 & +1 \\ -1 & 0 \end{pmatrix} \right\}^n \quad (7)$$

für  $n \in \{100, 1000, 10000, 100000\}$  und  $t \in [0, T] = [0, 15]$ , wählen Sie hier  $dt = 0.1$ , und plotten Sie die Grössen

$$x(t) := E_n(t)_{1,1} \quad (8)$$

$$y(t) := E_n(t)_{1,2} \quad (9)$$

in der  $(x, y)$ -Ebene. Dabei sei  $Id$  die  $2 \times 2$  Einheitsmatrix.

e) Offensichtlich können wir das Matrix-Exponential  $e^A$  näherungsweise durch den Ausdruck

$$e^A \stackrel{n \text{ gross}}{\approx} \left\{ Id + \frac{A}{n} \right\}^n \quad (10)$$

berechnen wenn wir das  $n$  hinreichend gross wählen. Wenn wir nun für eine beliebige Matrix  $B$  (das  $B$  wird dann das  $Id + A/n$ ) die Matrizen

$$\begin{aligned} B_0 &:= B \\ B_1 &:= B_0 B_0 \\ B_2 &:= B_1 B_1 \\ &\vdots \\ B_m &:= B_{m-1} B_{m-1} \end{aligned} \quad (11)$$

mit Hilfe von  $m$  Matrixprodukten berechnen, dann ist offensichtlich

$$B_m = B^n = B^{2^m}, \quad (12)$$

wir können also etwa  $B^{1024}$  mit nur 10 Matrix-Multiplikationen berechnen. Codieren Sie jetzt eine Funktion für ein effizientes Matrix-Exponential.