
Von der mathematischen Biologie zur Systembiologie

(Vorlesung Prof. Dr. J. Timmer)

Aufgabenzettel Nr. 0

Vorbereitung auf die Übungen (wichtig):

Sie werden in dieser Vorlesung unter anderem lernen mit der Programmiersprache R (siehe <http://cran.r-project.org/>) zu arbeiten. Der Einstieg in die Übungen wird Ihnen leicht(er) fallen, wenn Sie sich bereits vor der ersten Übungsstunde mit den einfachen Strukturen von R auseinandergesetzt haben. Machen Sie sich daher mit den Kapiteln 1-3,5,9,10 der Einführung „An Introduction To R“ vertraut, die Sie hier <http://cran.r-project.org/manuals.html> finden.

Präsenzaufgabe -3: Vektoren in R

- i.) Generieren Sie die folgenden Vektoren. Dabei sind die vordefinierten Vektoren `LETTERS` und `letters` ggf. hilfreich, sowie z.B. die Funktionen `c()`, `rep()`, `paste()`, `seq()`, `as.character()`, `as.integer()` und `:`.
 - (a) 1, 2, 3, 4, ..., 100
 - (b) "1", "2", "3", "4", ..., "100"
 - (c) "A1a", "B2b", ..., "Z26z"
 - (d) 0.2, 0.4, 0.6, ..., 20
 - (e) TRUE, TRUE, FALSE, FALSE, TRUE, TRUE, FALSE, FALSE, TRUE, TRUE, FALSE, FALSE
 - (f) "A", "B", "C", "A", "B", "C", "A", "B", "C", "A", "B", "C", "A", "B", "C"
- ii.) Von welchem *type* und welchem *mode* sind die Vektoren in i.)?
- iii.) Für welche der obigen Vektoren sind die unären Operatoren `+`, `-`, `!` (sinnvoll) anwendbar?
- iv.) Auf welche der obigen Vektoren kann man die Funktionen `as.character()` und `as.integer()` (sinnvoll) anwenden?

Präsenzaufgabe -2: Einfache und nützliche Funktionen

Mit dieser Aufgabe sollen ein paar weitere nützliche Funktionen zur Manipulation von Vektoren und Matrizen angewendet werden: zum Beispiel `dim()`, `length()`, `matrix()`, `diag()`, `t()`, `outer()`, `sum()`, sowie die binären Operatoren `*`, `/`, `+`, `-`, `^` und der `[]`-Operator.

- i.) Erstellen Sie eine 10×10 -Matrix `A`, deren Einträge das kleine 1×1 ist. Summieren Sie nun alle Einträge.
- ii.) Erstellen Sie eine 26×26 -Matrix `C`, wobei in jeder Zeile (bei `A` anfangend) alle Großbuchstaben stehen. Es ist verboten beim Erstellen der Matrix Zahlen zu verwenden!
- iii.) Ändern Sie die Matrix `C` derart, dass aus den "V"-Einträgen alle Quadratzahlen von "1" bis "676" werden.
- iv.) Schränken Sie nun die neu entstandenen Matrix `C` so ein, dass sie immer noch quadratisch ist, aber nur noch die Buchstaben E, F, G, H, I und W, X, Y, Z vorkommen. Die neue Matrix ist also eine 9×9 -Matrix.
- v.) Erstellen Sie mit Hilfe der neuen 9×9 -Matrix `C` eine Matrix `E`, deren Einträge `TRUE` sind für alle Buchstaben im Alphabet vor K und `FALSE` für alle anderen.

vi.) Was ergeben `length(C)` und `dim(C)`?

vii.) Generieren Sie nun mit `diag()` aus der Matrix `C` eine neue Matrix `F`, deren Diagonale mit der von `E` identisch ist und deren restliche Einträge "0" sind. Was fällt auf?

Präsenzaufgabe -1: Lineares Gleichungssystem in R

Gegeben sei das lineare Gleichungssystem $a = Mx$ mit

$$a = \begin{pmatrix} 1 \\ 2 \\ 4 \end{pmatrix}, \quad \text{und} \quad M = \begin{pmatrix} 2 & 3 & 4 \\ 3 & 4 & 1 \\ 4 & 5 & 6 \end{pmatrix}. \quad (1)$$

- i.) Lösen Sie mit Hilfe von R das obige lineare Gleichungssystem. Benutzen Sie zur Eingabe der Matrix die Funktionen `c()` und `dim()`.
- ii.) Bestimmen Sie die Inverse M^{-1} von M sowie die Eigenwerte und die Eigenvektoren von M .
- iii.) Bestimmen Sie mit Hilfe des binären Matrixprodukts `%*%` die quadratische Form $q = x^T M^{-1} x$.
- iv.) Machen Sie nun (mit dem `[-`Operator) die Änderung $M_{23} = 5$, und lösen erneut das Gleichungssystem. Was ändert sich an der Ausgabe und warum?

Präsenzaufgabe 0: Schleifen und Funktionen in R

i.) Die *Collatz Vermutung*:

Man bildet eine Zahlenfolge $(a_n)_{n \in \mathbb{N}}$ nach dem folgenden Schema: man startet mit einer beliebigen natürlichen Zahl $a_1 \geq 1$. Jedes Folgeelement a_{n+1} wird wie folgt gebildet:

- Ist a_n gerade, so ist $a_{n+1} = \frac{1}{2}a_n$.
- Ist a_n ungerade, so ist $a_{n+1} = 3a_n + 1$.

Die Collatz-Vermutung sagt nun, dass – unabhängig vom Startwert – jede Folge endliche Länge hat (es also keine Schleifen gibt) und mit den Elementen $a_{\ell-2} = 4$, $a_{\ell-1} = 2$ und $a_\ell = 1$ endet.

Aufgabe: Schreiben Sie eine Funktion `collatz()`, deren Argument eine natürliche Zahl (ein Integer) ist, und deren Rückgabewert eine Liste ist. Diese soll zum einen den Vektor mit allen Folgegliedern enthalten und zum anderen die Anzahl ℓ der Folgeglieder.

Tipp: Der Divisionsalgorithmus (*modulo* Rechenoperation) ist der binäre Operator `%`. Weitere nützliche Funktionen sind ggf. `all.equal()` und `isTRUE()`.

ii.) Schreiben Sie eine Funktion `prime()`, die als Argument eine natürliche Zahl n (einen Integer) hat, und deren Rückgabewert ein Vektor mit den ersten n Primzahlen ist.