

Binäre Bäume mit Bewertung

- ▶ A totalgeordnete Menge (z.B. $\mathbb{Z}, \mathbb{R}, \dots$)
- ▶ Für $t = \langle \circ, t_\ell, t_r \rangle \in \mathcal{B}$ und $v : I(t) \rightarrow A$ sei

$$v_\ell : I(t_\ell) \rightarrow A : a \mapsto v(a)$$

$$v_r : I(t_r) \rightarrow A : a \mapsto v(a)$$

die Restriktion von v auf den linken bzw. rechten Teilbaum

Binäre Suchbäume

- ▶ Ein Paar (t, v) mit $t \in \mathcal{B}, v : I(t) \rightarrow A$ heisst *binärer Suchbaum* über A , wenn
 - ▶ $t = \square$ (und somit $v = \lambda =$ leere Funktion), oder
 - ▶ $t = \langle \circ, t_\ell, t_r \rangle$ mit $v(\circ) = a$ und
 1. (t_ℓ, v_ℓ) ist ein binärer Suchbaum über $A_{<a} = \{b \in A; b < a\}$
 2. (t_r, v_r) ist ein binärer Suchbaum über $A_{>a} = \{b \in A; b > a\}$

Insbesondere gilt also

$$\max_{i \in I(t_\ell)} v(i) < v(\circ) = a < \min_{i \in I(t_r)} v(i)$$

- ▶ $\mathcal{BS}(A)$: Menge der binären Suchbäume über A
- ▶ $\mathcal{BS}_n(A)$: Menge der $(t, v) \in \mathcal{BS}(A)$ mit $t \in \mathcal{B}_n$

- ▶ Struktur:

$$\mathcal{BS}(A) = \bigsqcup_{a \in A} (\{a\} \times \mathcal{BS}(A_{<a}) \times \mathcal{BS}(A_{>a}))$$

- ▶ Anzahl:

$$\#\mathcal{BS}_n(A) = \binom{\#A}{n} \cdot c_n = \binom{\#A}{n} \cdot \frac{1}{n+1} \binom{2n}{n}$$

Suchen in binären Suchbäumen

Suche, ob ein gegebenes $a \in A$ in $(t, v) \in \mathcal{BS}(A)$ vorkommt:

- ▶ falls $t = \square$: FAIL
- ▶ falls $t = \langle \circ, t_\ell, t_r \rangle$
 - ▶ falls $v(\circ) = a$: FOUND
 - ▶ falls $v(\circ) > a$: suche a in (t_ℓ, v_ℓ)
 - ▶ falls $v(\circ) < a$: suche a in (t_r, v_r)
- ▶ Die (mittlere) innere Weglänge von t ist ein Maß für den (mittleren) Aufwand erfolgreicher Suche in t .
- ▶ Die (mittlere) äussere Weglänge von t ist ein Maß für den (mittleren) Aufwand erfolgloser Suche in t .

Rekursive Konstruktion von binären Suchbäumen

- Für eine Folge $\mathbf{a} = a_1 a_2 \dots a_n \in A^+$ sei

$\mathbf{a}_{<}$: die Teilfolge der $a_i < a_1$,

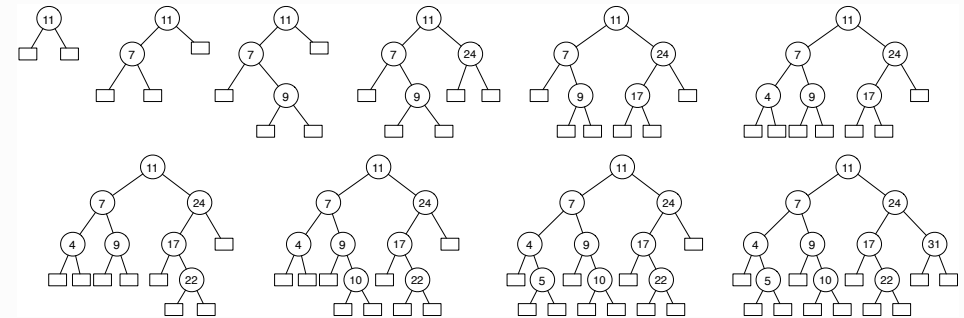
$\mathbf{a}_{>}$: die Teilfolge der $a_i > a_1$

- Für die leere Folge λ sei $bs(\lambda) = (\square, \lambda)$
- Für $a \in A^+$ wird $bs(\mathbf{a}) = (t, v)$ definiert durch

$$v(\circ) = a_1, \quad bs(\mathbf{a}_{<}) = (t_\ell, v_\ell), \quad bs(\mathbf{a}_{>}) = (t_r, v_r)$$

- Ein sequentieller Algorithmus zur Konstruktion verwendet sukzessives Einfügen bei erfolgloser Suche

Konstruktion des binären Suchbaumes zu
 $\mathbf{a} = (11, 7, 9, 24, 17, 4, 22, 10, 5, 31)$

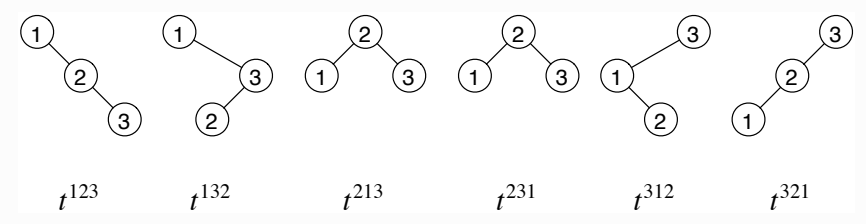


- Problem: wie gross ist der Aufwand für (erfolgreiche) Suche, wenn man aus Daten sequentiell einen binären Suchbaum konstruiert hat?
- Wird $a \in A$ in den A -Suchbaum (t, v) (erfolgreich) gesucht, so ist die Höhe $h(b, t)$ des Knotens $b \in I(t)$ mit $v(b) = a$ relevant.
- Damit ist klar: es kommt sowohl auf das gesuchte Datum an, als auch auf den Suchbaum (d.h. die Reihenfolge, in der die Daten zur Konstruktion des Suchbaumes verwendet wurden).

Permutationsmodell

- Für n (verschiedene) Daten $\in A$ betrachte alle $n!$ Permutationen als gleichwahrscheinliche inputs für die Konstruktion eines bs .
- Man kann annehmen, dass $A = \{1, 2, \dots, n\}$ und alle $\sigma \in \mathcal{S}_n$ inputs sind. Für $\sigma \in \mathcal{S}_n$ sei $bs(\sigma) = (t^\sigma, v^\sigma)$, also $t \in \mathcal{B}_n$.
 - Vorsicht: verschiedene $\sigma \in \mathcal{S}_n$ können denselben Suchbaum liefern!
 - v^σ ist durch t^σ eindeutig bestimmt, also redundant!
- Es wird mit gleicher Wahrscheinlichkeit nach jedem der Daten $k \in \{1, 2, \dots, n\}$ gesucht
- Für einen festen A -Suchbaum (t, v) ist die (mittlere) innere Weglänge $w_i(t)$ bzw. $\bar{w}(t) = w_i(t)/n$ das Maß für den Suchaufwand.

Beispiel: die Suchbäume zu den Permutationen aus \mathcal{S}_3



Es gilt

$$w_i(t^{123}) = w_i(t^{132}) = w_i(t^{312}) = w_i(t^{321}) = 3, w_i(t^{213}) = w_i(t^{231}) = 2$$

also

$$w_3 = \frac{1}{3!} \sum_{\sigma \in \mathcal{S}_3} w_i(\sigma) = \frac{3 + 3 + 2 + 2 + 3 + 3}{6} = \frac{8}{3}, \bar{w}_3 = \frac{1}{3} w_3 = \frac{8}{9}$$

► Gesucht ist

$$w_n = \frac{1}{n!} \sum_{\sigma \in \mathcal{S}_n} w_i(t^\sigma) \text{ bzw. } \bar{w}_n = \frac{1}{n!} \sum_{\sigma \in \mathcal{S}_n} \bar{w}(t^\sigma) = \frac{1}{n} w_n$$

► Wir werden sehen: die w_n erfüllen die Quicksort-Rekursion!

$$w_n = \frac{2}{n} \sum_{k=1}^{n-1} w_k + n - 1, w_0 = 0$$

► Folgerung: der mittlere Suchaufwand bei binären Suchbäumen im Permutationsmodell ist

$$\bar{w}_n = \frac{1}{n} w_n = (2 + \frac{2}{n}) H_n - 4 \in \Theta(\log n)$$

► Wie oft tritt jeder binäre Baum $t \in \mathcal{B}_n$ bei der Konstruktion von binären Suchbäumen auf?

$$s_t = \#\{\sigma \in \mathcal{S}_n; t^\sigma = t\}$$

► Behauptung:

$$s_t = \frac{n!}{\prod_{a \in I(t)} i(t_a)}$$

Dabei bezeichnet t_a den Teilbaum von t mit Wurzel a

► Beispiel: $\sigma = [5, 1, 3, 2, 7, 6, 4] \in \mathcal{S}_7$ erzeugt $t^\sigma \in \mathcal{B}_7$ mit $\text{code}(t^\sigma) = \bigcirc \bigcirc \square \bigcirc \bigcirc \square \square \bigcirc \square \square \bigcirc \bigcirc \square \square \square$
Identifiziert man die Zahlen $a \in \{1, \dots, 7\}$ mit den Knoten, an denen sie in t^σ stehen, so gilt
 $i(t_1) = 4, i(t_2) = 1, i(t_3) = 3, i(t_4) = 1, i(t_5) = 7, i(t_6) = 1$
und $i(t_7) = 2$ und es gibt $s_{t^\sigma} = \frac{7!}{4 \cdot 1 \cdot 3 \cdot 1 \cdot 7 \cdot 1 \cdot 2} = 30$
Permutationen aus \mathcal{S}_7 , die den gleichen binären Baum erzeugen.

Zum Beweis:

► Betrachte die Abbildung

$$\mathcal{S}_n \ni \sigma \mapsto (P_\sigma, \sigma_<, \sigma_>^{\ominus \sigma_1}) \in \binom{2..n}{\sigma_1-1} \times \mathcal{S}_{\sigma_1-1} \times \mathcal{S}_{n-\sigma_1}$$

Dabei ist (mit $\sigma_1 = k$):

- $P_\sigma = \{i_1, i_2, \dots, i_{k-1}\} \subseteq \binom{2..n}{k-1}$ mit $2 \leq i_1 < i_2 < \dots < i_{k-1} \leq n$ die Positionen $2 \leq i \leq n$ mit $\sigma_i < k$
- $\sigma_< = \sigma_{i_1} \sigma_{i_2} \dots \sigma_{i_{k-1}} \in \mathcal{S}_{k-1}$
- $\sigma_>^{\ominus k} = (\sigma_{j_1} - k)(\sigma_{j_2} - k) \dots (\sigma_{j_{n-k}} - k) \in \mathcal{S}_{n-k}$, wobei $j_1 < j_2 < \dots < j_{n-k} \leq n$ die Positionen $2 \leq j \neq n$ mit $\sigma_j > k$ sind

Beispiel

$$\mathcal{S}_n \ni \sigma \mapsto (P_\sigma, \sigma_<, \sigma_>^{\ominus \sigma_1}) \in \binom{2..n}{\sigma_1-1} \times \mathcal{S}_{\sigma_1-1} \times \mathcal{S}_{n-\sigma_1}$$

mit $n = 9$

- ▶ $\sigma = (681379542)$, also $k = \sigma_1 = 6$
- ▶ $P_\sigma = (34789) \in \binom{2..9}{5}$
- ▶ $\sigma_< = (13542) \in \mathcal{S}_5$
- ▶ $\sigma_>^{\ominus 6} = (213) \in \mathcal{S}_3$

▶ Diese Abbildung

$$\sigma \mapsto (P_\sigma, \sigma_<, \sigma_>^{\ominus \sigma_1}) : \mathcal{S}_n \rightarrow \bigsqcup_{1 \leq k \leq n} \binom{2..n}{k-1} \times \mathcal{S}_{k-1} \times \mathcal{S}_{n-k}$$

ist eine Bijektion!

▶ Es gilt für alle $\sigma \in \mathcal{S}_n$ und mit $\sigma_\ell = \sigma_<$, $\sigma_r = \sigma_>^{\ominus \sigma_1}$

$$t^\sigma = \langle \circlearrowleft, t^{\sigma_\ell}, t^{\sigma_r} \rangle$$

▶ Damit ist per Induktion für $t \in \mathcal{B}_n$ mit $t_\ell \in \mathcal{B}_{k-1}$, $t_r \in \mathcal{B}_{n-k}$

$$\begin{aligned} s_t &= \#\{\sigma \in \mathcal{S}_n; t^\sigma = t\} = \binom{n-1}{k-1} \cdot s_{t_\ell} \cdot s_{t_r} \\ &= \frac{(n-1)!}{(k-1)!(n-k)!} \cdot \frac{(k-1)!}{\prod_{a \in I(t_\ell)} i((t_\ell)_a)} \cdot \frac{(n-k)!}{\prod_{b \in I(t_r)} i((t_r)_b)} \\ &= \frac{n!}{\prod_{a \in I(t)} i(t_a)} \end{aligned}$$

▶ Mit

$$\mathbf{p}_n : \mathcal{B}_n \mapsto [0, 1] : t \mapsto \mathbf{p}_n(t) = \frac{\#\{\sigma \in \mathcal{S}_n; t^\sigma = t\}}{n!} = \frac{1}{\prod_{a \in I(t)} i(t_a)}$$

wird eine Wahrscheinlichkeitsverteilung \mathbf{p}_n auf \mathcal{B}_n definiert: sie gibt für jeden binären Baum $t \in \mathcal{B}_n$ die Wahrscheinlichkeit an, mit welcher er bei der Konstruktion binärer Suchbäume unter dem Permutationsmodell auftritt.

▶ Es gilt für $t \in \mathcal{B}_n$ mit $t_\ell \in \mathcal{B}_{k-1}$, $t_r \in \mathcal{B}_{n-k}$

$$\mathbf{p}_n(t) = \frac{1}{n} \cdot \mathbf{p}_{k-1}(t_\ell) \cdot \mathbf{p}_{n-k}(t_r)$$

▶ Beweis der Quicksort-Rekursion für die mittlere Weglänge

$$\begin{aligned} w_n &= \frac{1}{n!} \sum_{\sigma \in \mathcal{S}_n} w_i(t^\sigma) = \sum_{t \in \mathcal{B}_n} \mathbf{p}_n(t) \cdot w_i(t) \\ &= \sum_{k=1}^n \sum_{\substack{t_\ell \in \mathcal{B}_{k-1} \\ t_r \in \mathcal{B}_{n-k}}} \frac{1}{n} \cdot \mathbf{p}_{k-1}(t_\ell) \cdot \mathbf{p}_{n-k}(t_r) (w_i(t_\ell) + w_i(t_r) + n) \\ &= \frac{1}{n} \sum_{k=1}^n (w_{k-1} + w_{n-k} + n - 1) = \frac{2}{n} \sum_{k=0}^{n-1} w_k + n - 1 \end{aligned}$$