KAPITEL 2 : STANDORTPROBLEME

2.1 Einstieg

Betrachte: n Punkte auf einer Geraden, beschrieben durch $x_1, \ldots, x_n$

Wir wollen jetzt besondere Eigenschaften und Positionen bestimmen:

- den Punkt mit kleinster oder größter Koordinate
- einen „mittleren" Punkt (oder Nr. k unter den Punkten)
- Punkte ~~aus~~ einem bestimmten Intervall
- „Lücken" in der Anordnung

Dabei entwickeln wir auch einen genaueren Blick für

- bestimmte algorithmische Konzepte (Preprocessing <-> Query)
- Anwendungen für Rekursionen
- raffiniertere Datenstrukturen

Beachte: - n kann sehr groß sein!
- Die Punkte müssen nicht sortiert sein!

Problem 2.1 (Minimum, Maximum)

Gegeben: $x_1, \dots, x_n$

Gesucht: i^* und j^* mit $x_{i^*} = \min\{x_i \mid i \in \{1,\dots,n\}\}$

$x_{j^*} = \max\{x_j \mid j \in \{1,\dots,n\}\}$.

Beobachtungen:

- Schneller als in $\Omega(n)$ geht das nicht: Wir müssen alle x_i anschauen.
- Das geht sehr leicht in $O(n)$: Gehe einmal alle Zahlen durch und merke jeweils das kleinste bzw. größte Element.

Problem 2.2

Gegeben: n Punkte auf einer Geraden, beschrieben (o.B.d.A.) durch $x_1, \dots, x_n$

Gesucht: Ein Punkt m, so dass

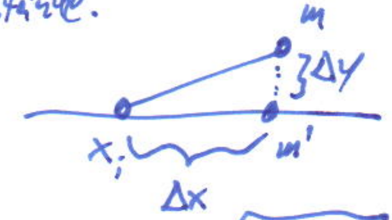
$$\sum_{i=1}^{n} d(m, x_i)$$

kleinstmöglich ist.

Beobachtungen:

(1) m sollte auch auf der Geraden liegen.

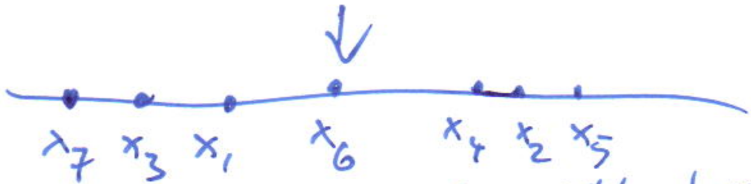
(Sonst kann man den Punkt auf die Gerade projizieren und bekommt kleinere Abstände.



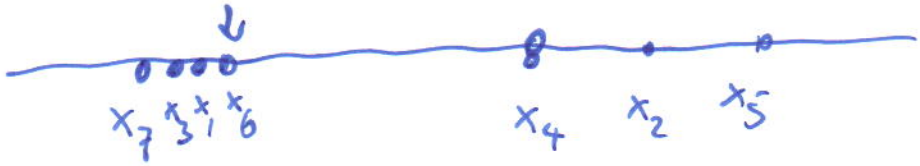
$$d(m, x_i) = \sqrt{\Delta x^2 + \Delta y^2},$$

also $d(m', x_i) = \Delta x \le d(m, x_i)$)

(2) Unter den Punkten auf der Geraden sollte man m so wählen, dass er „in der Mitte" sitzt:



Selbst wenn das nicht der Mittelwert ist!



(3) Der Schwerpunkt kann auch nützlich sein, aber er minimiert nicht

$$\sum_{i=1}^{n} d(m, x_i),$$

sondern

$$\sum_{i=1}^{n} d(m, x_i)^2$$

) !

Also:

DEFINITION 2.3 (Median)

(1) Für n Zahlen $x_1, \ldots, x_n$ ist ein <u>Median</u> eine Zahl m mit den Eigenschaften

$$|\{x_i \mid x_i < m\}| \leq \frac{n}{2}$$

$$|\{x_i \mid x_i > m\}| \geq \frac{n}{2}.$$

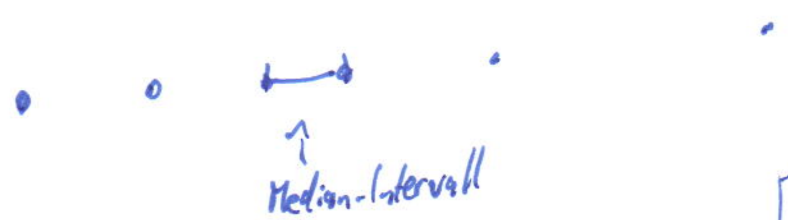
(4) Das lässt sich auch für Verteilungen beschreiben:

Mit $\int_{-\infty}^{\infty} f(x)\,dx =: A$ ist m Median, wenn

$$\int_{-\infty}^{m} f(x)\,dx \leq \frac{A}{2}$$

$$\int_{m}^{\infty} f(x)\,dx \leq \frac{A}{2}$$

(2) Für ungerades n ist ein Median eindeutig, für gerades n kann (!) es mehr als einen geben:



(3) Man kann das verallgemeinern: m hat Rang k in X, wenn

$$|\{x_i \mid x_i \leq m\}| \geq k$$

$$|\{x_i \mid x_i \geq m\}| \geq n-k$$

Bem.: Median gut für viele Zwecke!

2.2 Medianberechnung

PROBLEM 2.4 (Median)

Gegeben: n Zahlen $\{x_1, \dots, x_n\} =: X$

Gesucht: Ein Median m von X

Allgemeiner:

PROBLEM 2.5 (Rang-k-Element)

Gegeben: n Zahlen $\{x_1, \dots, x_n\} =: X$, $k \in \{1, \dots, n\}$

Gesucht: Ein Element von Rang k in X

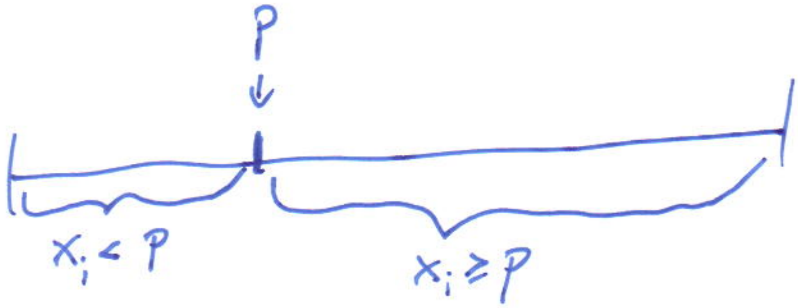
Beobachtung:

- Man kann beide Probleme in $O(n \log n)$ lösen, indem man X sortiert und danach abzählt.

Geht das schneller?!

Idee dafür inspiriert von QUICKSORT!

- Wähle ein Pivotelement ← Wie?!
- Teile die Menge in größere und kleinere Elemente



- Bei QUICKSORT: Beide Teile rekursiv weiter unterteilen
- Hier: Nur ein Teil wird weiter unterteilt – derjenige, der den Median (oder allgemeiner: Element mit Rang k) enthält.

- Aber: Wenn man den Pivot schlecht wählt, dann dauert es trotzdem lange – wie beim Worst Case von QUICKSORT kann das $\Omega(n^2)$ sein.

- Bessere Auswahl des Pivotelements?!

  Ziel: Vermeide Worst Case, indem jeweils ein Minimal- und Maximalanteil kleiner und größer ist.