

1. Übung Informationstheorie SS20

1. Kanonische Codes: je nachdem, wie der Fragebaum gebaut wird, kommt man zu unterschiedlichen Codewörtern für den Huffman-Code. Um Eindeutigkeit zu erreichen, kann man wie folgt vorgehen (dieses Verfahren lässt sich auf jeden Präfixfreien Code anwenden):

Zuerst werden mit dem Huffman-Algorithmus nur die Codewortlängen bestimmt. Diese werden ansteigend geordnet:

$$l_1 \leq l_2 \leq \dots \leq l_m.$$

Die Codewörter (als Binärzahl mit l_i Stellen gelesen, wenn nötig mit führenden Nullen) erhält man durch

$$c_1 = 0,$$

$$c_{i+1} = (c_i + 1)2^{l_{i+1}-l_i}, i = 1, \dots, m - 1.$$

Zeigen Sie dass dadurch ein präfixfreier Code erzeugt wird (dieser Code wird der kanonische Code genannt. Ein Vorteil dieses Verfahrens ist, dass zur Übertragung des Codes nur die Codewortlängen angegeben werden müssen).

2. Schreiben Sie (in einer Programmiersprache Ihrer Wahl) ein Programm zur Bestimmung eines Huffman-Codes und seiner kanonischen Form. Überlegen Sie, wie (ob,dass) das (teure) Sortieren der Wahrscheinlichkeiten in jedem Schritt vereinfacht/vermieden werden kann.
3. Bestimmen Sie die Entropie der Verteilung $P = (0.1, 0.2, 0.3, 0.4)$ und vergleichen Sie sie mit der mittleren Unbestimmtheit.
4. Bestimmen Sie für $m = 1, 2, 3, 4$ explizite Ausdrücke für $H^*(P)$ (Nehmen Sie an, dass die Wahrscheinlichkeiten absteigend geordnet sind).
5. Zeigen Sie, dass $H^*(P) = H(P)$ genau dann gilt, wenn alle p_i von der Form 2^{-k} sind. Bestimmen sie alle solchen Verteilungen für $m = 1, 2, 3, 4$.
6. Gegeben Sei die Verteilung

$$P = (0.1, 0.4, 0.05, 0.2, 0.25)$$

Bestimmen sie die mittlere Unbestimmtheit und die Entropie.

7. Angesichts der Darstellung der Entropie und ihrer Nähe zur mittleren Unbestimmtheit liegt die Vermutung nahe (und ist als heuristisches Argument für einige Ergebnisse auch recht brauchbar), dass in einem optimalen Code $l_i \approx \log_2(1/p_i)$ gelten sollte. In der einen Richtung kann das dramatisch falsch sein, wie das Beispiel $m = 2, p_1$ klein zeigt: hier ist $l_1 = 1$ sehr viel kleiner als $\log_2(1/p_1)$. In der anderen Richtung sieht die Sache besser aus:

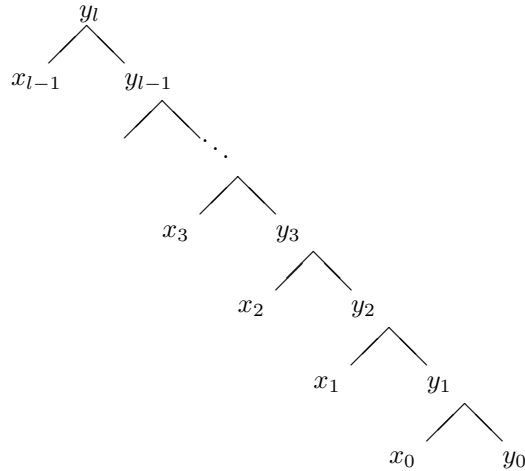
(f_n) seien die Fibonaccizahlen ($f_0 = 0, f_1 = 1, f_{n+1} = f_n + f_{n-1}$).

Zeigen Sie für $i \geq 1$

$$f_i \geq \tau^{i-2},$$

wobei $\tau = \frac{1+\sqrt{5}}{2}$ die positive Lösung von $\tau^2 = \tau + 1$ ist.

In einem Huffmanbaum betrachten wir einen Pfad der Länge l :



x_i und y_i repräsentieren die Wahrscheinlichkeiten der Knoten.

Zeigen Sie, dass $x_0 \geq 0$, $y_i = x_{i-1} + y_{i-1}$ und $x_i \geq y_{i-1}$ gilt und dass

$$y_i \geq f_{i+1} y_0.$$

Folgern Sie daraus

$$l \leq \log_\tau\left(\frac{1}{y_0}\right) + 1 \leq 1.5 \log_2\left(\frac{1}{y_0}\right) + 1.$$