

Quellencodierung I:

Redundanzreduktion, redundanzsparende Codes

1. Redundanz

1.1 Einführung

1.2 Definition der Redundanz

1.3 allgemeine Redundanzreduktion

2. redundanzsparende Codes

2.1 Codierung nach Shannon

2.2 Codierung nach Fano

2.3 Codierung nach Huffman

2.4 Codierung von Zeichenfolgen

2.5 Redundanzreduktion durch Codeumschaltung

1. Redundanz

1.1 Einführung

Redundanz: Bezeichnung für die Anteile einer Nachricht, die keine Information vermitteln, also überflüssig sind... .

(aus: Duden Informatik, Dudenverlag, Mannheim 1993)

Beispiel: BCD-Code

Zeichen	Codewort
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Der Code ist redundant, weil nur 10 der 16 möglichen Codewörter benutzt werden.

*

Beispiel: ASCII-Code

Zeichen	Codewort
NUL	0000000
SOH	0000001
STX	0000010
...	...
x	1111000
y	1111001
z	1111010
{	1111011
	1111100
}	1111101
~	1111110
DEL	1111111

Alle 128 möglichen Codewörter werden genutzt (Minimalcode). Trotzdem ist der Code redundant, denn die Zeichen treten mit unterschiedlichen Wahrscheinlichkeiten auf.

*

1.2 Definition der Redundanz

Definition: Die **Codewortlänge** eines Codewortes ist die Anzahl der Binärzeichen (0 oder 1), die bei der Codierung des Codewortes benutzt werden.

Definition: Die **mittlere Codewortlänge** m ist der gewichtete Mittelwert (d.h. der Erwartungswert) der Codewortlängen aller n Zeichen:

$$m := \sum_{i=1}^n p(x_i) \cdot m_i$$

mit $p(x_i)$: Wahrscheinlichkeit, daß das i -te Zeichen auftritt,
 m_i : Codewortlänge des i -ten Zeichens.

Definition: Der **mittlere Informationsgehalt** H von n Zeichen ist der gewichtete Mittelwert des Informationsgehaltes I_i der einzelnen Zeichen:

$$H := \sum_{i=1}^n p(x_i) \cdot I_i$$

mit $I_i = \log_2 \left(\frac{1}{p(x_i)} \right)$: Informationsgehalt des i -ten Zeichens,
 $p(x_i)$: Wahrscheinlichkeit, daß das i -te Zeichen auftritt.

Definition: Die **Redundanz** eines Codes ist die Differenz zwischen der mittleren Codewortlänge und dem mittleren Informationsgehalt:
 $R := m - H$.

Die **relative Redundanz** gibt an, wieviel Prozent der Codierung redundant sind:

$$r := \frac{m - H}{m}.$$

1.3 allgemeine Redundanzreduktion

- statt Blockcodes Codewörter unterschiedlicher Länge verwenden
- Zeichen mit hoher Wahrscheinlichkeit erhalten kurzes Codewort
- Zeichen mit geringer Wahrscheinlichkeit erhalten langes Codewort

Problem: Übermittlung der Codewortlänge

Beispiel:

Zeichen	Codewort
1	0
2	1
3	10

Zeichenfolge 1, 2, 3 wird codiert als 0110;

0110 ist nicht eindeutig decodierbar: entweder 1, 2, 3 oder 1, 2, 2, 1

*

Fano-Bedingung (Präfix-Eigenschaft):

Kein Codewort aus einem Code bildet den Anfang eines anderen Codewortes.

Shannonsches Codierungstheorem:

Die Codierung eines Zeichenvorrats kann immer so vorgenommen werden, daß die Redundanz minimal wird.

2. redundanzsparende Codes

2.1 Codierung nach Shannon

Voraussetzungen:

- n verschiedene Zeichen x_1, \dots, x_n mit Wahrscheinlichkeiten $p(x_1), \dots, p(x_n)$
- Zeichen sind nach fallender Wahrscheinlichkeit geordnet:
 $p(x_1) \geq p(x_2) \geq \dots \geq p(x_n)$.
- Für jedes Zeichen x_i ist P_i die kumulative Wahrscheinlichkeit aller vorigen Zeichen, also

$$P_1 = 0,$$

$$P_2 = p(x_1),$$

$$P_3 = p(x_1) + p(x_2),$$

$$P_4 = p(x_1) + p(x_2) + p(x_3),$$

...

$$P_n = p(x_1) + p(x_2) + \dots + p(x_{n-1}).$$

allgemein: $P_{i+1} = P_i + p(x_i)$ für $i = 1, \dots, n-1$

Berechnung der Codewortlänge:

Die zu benutzende Codewortlänge des i-ten Zeichens wird berechnet als

$$m_i = \lceil I_i \rceil = \left\lceil \log_2 \left(\frac{1}{p(x_i)} \right) \right\rceil.$$

Berechnung der Codewörter:

- kumulative Wahrscheinlichkeit P_i , die zum Zeichen x_i gehört, in eine Dualzahl umrechnen
- Vorkommastelle ignorieren, weil sie sowieso immer 0 ist
- Dualzahl nach der m_i -ten Stelle abbrechen (Codewort soll genau m_i Stellen lang sein)
- die Dualziffern ergeben das gesuchte Codewort

Beispiel:

Zeichen x_i	$p(x_i)$	P_i	m_i	Codewort	Z_i
A	0,4	0	2	00	0
B	0,2	0,4	3	011	0,375
C	0,15	0,6	3	100	0,5
D	0,15	0,75	3	110	0,75
E	0,05	0,9	5	11100	0,875
F	0,05	0,95	5	11110	0,9375

- mittlere Codewortlänge $m = 2,8$ Bit
- mittlerer Informationsgehalt $H = 2,246$ Bit
- Redundanz $R = 0,554$ Bit
- relative Redundanz $r = 19,77\%$

*

Beweis der Präfixeigenschaft:

Hilfssatz: Die Differenz zwischen zwei aufeinander folgenden kumulativen Wahrscheinlichkeiten ist mindestens 2^{-m_i} , d.h. $P_{i+1} - P_i \geq 2^{-m_i}$.

Beweis: Es ist $m_i = \left\lceil \log_2 \left(\frac{1}{p(x_i)} \right) \right\rceil$. Daraus folgt $m_i \geq \log_2 \left(\frac{1}{p(x_i)} \right)$.

Umformung nach $p(x_i)$ ergibt: $m_i \geq \log_2 \left(\frac{1}{p(x_i)} \right) \Leftrightarrow p(x_i) \geq 2^{-m_i}$.

Weil man die kumulative Wahrscheinlichkeit aus $P_{i+1} = P_i + p(x_i)$ berechnen kann, folgt $P_{i+1} - P_i \geq 2^{-m_i}$.

Der Hilfssatz besagt, daß *aufeinanderfolgende* Codewörter von der ersten bis zur m_i -ten Stelle verschieden sein müssen (Präfixeigenschaft).

Die Präfixeigenschaft zweier beliebiger Codewörter im Abstand n ergibt sich dadurch, daß man in der Formel $P_{i+1} - P_i \geq 2^{-m_i}$ die Variable i jeweils durch $i+1$, $i+2$, ..., $i+n-1$ ersetzt und alle Gleichungen, die man dabei erhält, addiert.

2.2 Codierung nach Fano

Voraussetzungen:

- n verschiedene Zeichen x_1, \dots, x_n mit Wahrscheinlichkeiten $p(x_1), \dots, p(x_n)$
- Zeichen sind nach fallender Wahrscheinlichkeit geordnet:
 $p(x_1) \geq p(x_2) \geq \dots \geq p(x_n)$.

Codierungsvorschrift:

1. Menge der Zeichen in 2 möglichst gleichwahrscheinliche Teile aufteilen
2. einer Teilmenge die Null als erste Codewortstelle zuordnen, der anderen die Eins
3. für beide Teilmengen diese Schritte rekursiv durchführen, bis einelementige Teilmengen entstehen

Beispiel:

Zeichen x_i	$p(x_i)$	Codewort
A	0,4	00
B	0,2	01
C	0,15	10
D	0,15	110
E	0,05	1110
F	0,05	1111

- mittlere Codewortlänge $m = 2,35$ Bit
- mittlerer Informationsgehalt $H = 2,246$ Bit
- Redundanz $R = 0,104$ Bit
- relative Redundanz $r = 4,41\%$

*

2.3 Codierung nach Huffman

Voraussetzungen:

- n verschiedene Zeichen x_1, \dots, x_n mit Wahrscheinlichkeiten $p(x_1), \dots, p(x_n)$
- Zeichen sind nach fallender Wahrscheinlichkeit geordnet:
 $p(x_1) \geq p(x_2) \geq \dots \geq p(x_n)$.

Codierungsvorschrift:

1. zwei Zeichen der kleinsten Wahrscheinlichkeit herausuchen
2. einem Zeichen die Null als (links anzufügende) Codewortstelle zuordnen, dem anderen die Eins
3. beide Zeichen zu einem zusammenfassen (Addition der Wahrscheinlichkeiten)
4. Verfahren wiederholen, bis alle Zeichen zusammengefaßt sind

Beispiel:

Schritt 1:

Zeichen x_i	$p(x_i)$	Codestelle
A	0,4	
B	0,2	
C	0,15	
D	0,15	
E	0,05	0
F	0,05	1

Schritt 2:

Zeichen x_i	$p(x_i)$	Codestelle
A	0,4	
B	0,2	
C	0,15	
D	0,15	0
E, F	0,1	1

Schritt 3:

Zeichen x_i	$p(x_i)$	Codestelle
A	0,4	
D, E, F	0,25	
B	0,2	0
C	0,15	1

Schritt 4:

Zeichen x_i	$p(x_i)$	Codestelle
A	0,4	
B, C	0,35	0
D, E, F	0,25	1

Schritt 5:

Zeichen x_i	$p(x_i)$	Codestelle
B, C, D, E, F	0,6	1
A	0,4	0

Zeichen x_i	$p(x_i)$	Codewort
A	0,4	0
B	0,2	100
C	0,15	101
D	0,15	110
E	0,05	1110
F	0,05	1111

- mittlere Codewortlänge $m = 2,3$ Bit
- mittlerer Informationsgehalt $H = 2,246$ Bit
- Redundanz $R = 0,054$ Bit
- relative Redundanz $r = 2,33\%$

*

2.4 Codierung von Zeichenfolgen

Hilfssatz: Beim Shannon-Code kann man die mittlere Codewortlänge abschätzen als $H \leq m < 1 + H$. Das bedeutet, daß die mittlere Codewortlänge m den mittleren Informationsgehalt um weniger als 1 Bit übersteigt.

Beweis: Die Codewortlänge beim Shannon-Code ist $m_i = \left\lceil \log_2 \left(\frac{1}{p(x_i)} \right) \right\rceil$.

Daraus folgen $m_i \geq \log_2 \left(\frac{1}{p(x_i)} \right)$ und $m_i < 1 + \log_2 \left(\frac{1}{p(x_i)} \right)$.

Die mittlere Codewortlänge m läßt sich damit auf zwei Arten abschätzen:

$$m = \sum_{i=1}^n p(x_i) \cdot m_i \geq \sum_{i=1}^n p(x_i) \cdot \log_2 \left(\frac{1}{p(x_i)} \right) = H \text{ und als zweites}$$

$$m = \sum_{i=1}^n p(x_i) \cdot m_i < \sum_{i=1}^n p(x_i) \cdot \left(1 + \log_2 \left(\frac{1}{p(x_i)} \right) \right) = \sum_{i=1}^n \left(p(x_i) + p(x_i) \cdot \log_2 \left(\frac{1}{p(x_i)} \right) \right)$$

$$= \sum_{i=1}^n p(x_i) + \sum_{i=1}^n p(x_i) \cdot \log_2 \left(\frac{1}{p(x_i)} \right) = 1 + H .$$

Man erhält also $H \leq m < 1 + H$.

2.4 Codierung von Zeichenfolgen

- Betrachtung von n verschiedenen Zeichen, die zu Zeichenfolgen der Länge N zusammengesetzt werden
- alle möglichen n^N Zeichenfolgen mit dem **Shannon-Code** codieren
- bei langen Zeichenfolgen verschwindet die Redundanz, d.h. $r \rightarrow 0$

Begründung:

- jedes Zeichen der Zeichenfolge habe mittleren Informationsgehalt H ;
also: mittlerer Informationsgehalt der Zeichenfolge $H_N = H \cdot N$,
- für die Zeichenfolge gilt die Abschätzung für die mittlere Codewortlänge
 $H_N \leq m < 1 + H_N$,
also folgt $H \cdot N \leq m < 1 + H \cdot N$,
- wenn die mittlere Codewortlänge nun nicht auf die Zeichenfolge, sondern auf die einzelnen Zeichen bezogen wird, ergibt sich $H \leq m' < \frac{1}{N} + H$ (mit $m' = \frac{m}{N}$);

daraus folgt $1 < \frac{1}{N \cdot m'} + \frac{H}{m'}$

- Abschätzung der relativen Redundanz:

$$r = \frac{m' - H}{m'} = 1 - \frac{H}{m'} < \frac{1}{N \cdot m'} + \frac{H}{m'} - \frac{H}{m'} = \frac{1}{N \cdot m'}$$

- weitere Abschätzung:

$$\text{aus } H \leq m' \Leftrightarrow \frac{1}{m'} \leq \frac{1}{H} \Leftrightarrow \frac{1}{N \cdot m'} \leq \frac{1}{N \cdot H} \text{ folgt } r < \frac{1}{N \cdot m'} \leq \frac{1}{N \cdot H}$$

- für wachsendes N strebt die relative Redundanz gegen Null,

$$\text{denn aus } 0 \leq r \leq \lim_{N \rightarrow \infty} \frac{1}{N \cdot H} = 0 \text{ folgt } r = 0.$$

Also: Bei der Codierung langer Zeichenfolgen strebt die Redundanz gegen Null.

2.5 Redundanzreduktion durch Codeumschaltung

Idee:

2 Codetabelle benutzen, zwischen denen mittels spezieller Codewörter umgeschaltet wird

Beispiel: Telegraphenalphabet

Buchstaben	Ziffern	Codewort
A	-	11000
B	?	10011
C	:	01110
D	wer da?	10010
E	3	10000
F	frei	10110
G	frei	01011
H	frei	00101
I	8	01100
J	BEL	11010
K	(11110
L)	01011
M	.	00111
N	,	00110
O	9	00011
P	0	01101
Q	1	11101
R	4	01010
S	‘	10100
T	5	00001
U	7	11100
V	=	01111
V	2	11001
X	/	10111
Y	6	10101
Z	+	10001
CR	CR	00010
LF	LF	01000
Buchstaben	Buchstaben	11111
Ziffern	Ziffern	11011
SP	SP	00100
		00000

*