Entropie en codering

Toegepaste Wiskunde 2007/2008

Het entropiebegrip van Shannon is een maat voor de hoeveelheid informatie die in een boodschap is vervat, en blijkt inderdaad asymptotisch samen te vallen met het minimale aantal bits dat voor verzending ervan nodig is. Dit verband krijgt een concrete betekenis in Huffman's codering, waarmee boodschappen efficient kunnen worden gecodeerd en computerbestanden gecomprimeerd.

1. Entropie als maat voor informatie

Stel je voor dat je, na afloop van een afmattende televisiequizz, wordt geconfronteerd met een glanzende ladenkast, en er wordt je verteld dat één van de laatjes je prijs bevat. Het is een vierkante kast met 16 laatjes. Door het stellen van janee-vragen moet je proberen de prijs te localiseren. Elke vraag kost je een vast bedrag. Je probeert daarom een vraagstrategie te bedenken, die het verwachte aantal benodigde vragen minimaal maakt. Wat wordt je strategie?

Als we aannemen dat elke laatje met dezelfde kans de prijs bevat, dan ligt de volgende strategie voor de hand: deel telkens de verzameling van nog mogelijkerwijs de prijs bevattende laatjes in tweeën, en vraag in welke helft zich de prijs bevindt. Je hebt dan met zekerheid na 4 vragen de prijs gelocaliseerd. Een andere strategie, die er duidelijk minder slim uitziet, zou zijn om de laatjes één voor één af te gaan, en telkens te vragen: 'Is het dit laatje?', 'Is het dit laatje?',.... Als je veel geluk hebt, vind je de prijs met deze strategie sneller, maar gemiddeld over de 16 mogelijke locaties van de prijs is het aantal benodigde vragen hoger (namelijk $8\frac{7}{16}$). We zullen zien dat de eerstgenoemde strategie in deze zin inderdaad optimaal is.

Het antwoord kan heel anders worden wanneer de kansverdeling die je veronderstelt voor de prijs er anders uitziet. Wanneer deze kansverdeling niet $(\frac{1}{16}, \frac{1}{16}, \frac{1}{16}, \ldots, \frac{1}{16})$ is, zoals hierboven verondersteld, maar bijvoorbeeld

$$(\tfrac{1}{2}, \tfrac{1}{4}, \tfrac{1}{8}, \ldots, 2^{-14}, 2^{-15}, 2^{-15}) \; .$$

dan is de boven beschreven strategie van één-voor-één vragen wèl optimaal. Over deze en dergelijke situaties bewees Shannon in 1948 zijn beroemde *broncoderingsstelling* (*"source coding theorem"*), ook wel bekend als *ruisvrije coderingsstelling* (*"noiseless coding theorem"*). We zullen met '$\log_2$' de logaritmische functie met grondtal 2 aanduiden.

Stelling 1. (Shannon) *Voor elke kansverdeling* $\mathbf{p} = (p_1, p_2, p_3, \ldots, p_n)$ *en elke*

vraagstrategie geldt:

$$\mathbb{E}(\text{aantal vragen}) \geq -\sum_{i=1}^{n} p_i \log_2 p_i \ . \tag{1}$$

Bovendien kan deze ondergrens willekeurig dicht benaderd worden als het spel vaak genoeg wordt gespeeld met onderling onafhankelijk geplaatste prijzen, en het toegestaan is vragen te stellen over combinaties van prijzen.

Het rechterlid in (1) wordt de *entropie* van de kansverdeling $\mathbf{p}$ genoemd, en genoteerd als $H(\mathbf{p})$.
We zullen in het onderstaande een bewijs geven van deze stelling, en vervolgens de optimale vraagstrategie beschrijven, die in 1952 door de MIT-promovendus Huffman is bedacht.

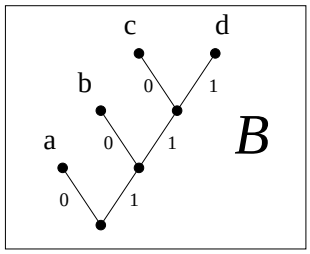
1.1 *Het verband met coderingstheorie*

Het spel met de prijs en de laatjes wordt beschreven met hetzelfde wiskundige model als het volgende *coderingsprobleem*: Hoe kun je een boodschap, bestaande uit een lange reeks letters, efficiënt coderen in 0-en en 1-en?
Het verband tussen beide wordt duidelijk als we de volgende 'vertaling' van het laatjesverhaal maken:

- De 'prijs' is een letter uit het alfabet.
- De 'vragen' zijn 0-en of 1-en die worden overgeseind om de letter aan te duiden. Een vraagstrategie is een codering van alle letters in rijtjes 0-en en 1-en.
- De kansen p_i zijn de relatieve frequenties waarmee de letters in gangbare boodschappen vóórkomen (of in de boodschap die voorligt).
- Een optimale vraagstrategie correspondeert dan met een codering waarmee boodschappen naar verwachting met zo weinig mogelijk bits kunnen worden overgestuurd. Betere coderingen worden mogelijk door blokken letters te coderen, in plaats van individuele letters.

2. Binaire bomen

Een vraagstrategie kan worden beschreven met behulp van een *binaire boom*. Daarom zullen we eerst wat moeten vertellen over boomstrukturen.
We beginnen met een voorbeeld.



Dit boompje beschrijft een vraagstrategie voor de prijzenkast met laatjes a, b, c en d. De strategie is tamelijk primitief en komt neer op: "Is het (niet) a?", "Is het (niet) b?", "Is het (niet) c?", "O, dan is het d!".
Het boompje kan ook worden gezien als de volgende codering van het "alfabet" $\{a, b, c, d\}$ in bitketens:

a	0
b	10
c	110
d	111

De code voor, bijvoorbeeld, de boodschap *abacadaba* is de bitketen

$$010011001110100\,.$$

Merk op dat er geen scheidingsteken nodig is, doordat geen enkele lettercode gelijk is aan een beginstuk van een andere. Bij het ontcijferen start je bij de wortel van de boom, en laat je door de bits naar boven leiden. Ben je bij een "blad" aangekomen, dan is je letter gevonden, en begin je weer onderaan. Het is bij deze codering daardoor van wezenlijk belang dat steeds wordt bijgehouden waar je zit in de boom, en op welke momenten opnieuw wordt begonnen bij de wortel.

Bij een boomje als dit hoort een natuurlijke kansverdeling, die ontstaat door, startend vanuit de wortel, bij elke vertakking een munt op te werpen, en aan de hand van de uitslag links- of rechtsaf te slaan. Bij dit boompje vind je de verdeling $(\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8})$ over het rijtje letters (a, b, c, d). In het geval dat deze kansverdeling overeenkomt met de frequenties van de letters in de tekst, dan is, zoals we zullen zien, de codering optimaal.

2.1 *Definities*

Een *boom* is een samenhangende graaf zonder cykels. Een *gewortelde boom* is een boom, waarin één knoop (vertex) is geselecteerd: de *wortel*, aan te duiden met $\odot$. Gewortelde bomen worden vaak getekend met de wortel onderaan. Van daaruit lopen zijden naar boven naar de buurknopen van de wortel, en van daaruit eventueel weer verder omhoog naar de buurknopen daarvan, etcetera. In analogie met stambomen worden hoger gelegen buurknopen ook wel *kinderen* van een vertex genoemd; de lager gelegen buurknoop heet de *ouder*. Vertices van waaruit geen zijden naar boven vetrekken worden de *bladeren* van de boom genoemd. De *hoogte* van een blad is het aantal zijden dat vanuit de wortel moet worden doorlopen om het blad te bereiken. De *hoogte* van de boom is die van het hoogste blad. (In de literatuur wordt het boven geschetste plaatje soms omgedraaid, zodat de wortel juist helemaal bovenaan afgebeeld staat.)
Een *binaire boom* is een gewortelde boom waarvan elke vertex 0, 1 of 2 kinderen heeft. Zo'n boom heet *volledig* als alleen de kindertallen 0 en 2 vóórkomen.
We kunnen van een gewortelde boom een nieuwe gewortelde boom maken door een blad te laten *ontkiemen*: we tekenen vanuit dit blad een aantal van $k \geq 1$ zijden

omhoog naar nieuwe bladeren $x_1, x_2, \ldots, x_k$. Het oude blad verdwijnt uit de lijst van bladeren, en $x_1, \ldots, x_k$ verschijnen. (Voor een binaire boom is $k = 1$ of 2; voor een volledige binaire boom is $k = 2$.)

Voor gewortelde bomen geldt een

Inductieprincipe. *Zij $P(B)$ een uitspraak over de gewortelde boom B. Stel dat geldt:*

i. $P(\{\odot\})$;

ii. *Voor elke gewortelde boom B en elke B' die daaruit is voortgekomen door ontkieming van een blad, geldt: $P(B) \Longrightarrow P(B')$.*

Dan is $P(B)$ juist voor elke gewortelde boom B.

Lemma 2. *Als $h_1, h_2, \ldots, h_n$ de hoogten zijn van de bladeren van een binaire boom B, dan geldt*

$$\sum_{j=1}^{n} 2^{-h_j} \leq 1 \ ,$$

met gelijkheid als de boom volledig is.

Bewijs. Noem het linkerlid van bovenstaande ongelijkheid: $S(B)$. Voor $B = \{\odot\}$ is $S(B) = 2^0 = 1$, want het enige blad is $\odot$ zelf, en dat zit op hoogte 0. Stel nu dat $S(B) \leq 1$ voor zekere B, en laat B' uit B ontstaan door ontkiemen van blad j in $k = 1$ of 2 hoger gelegen bladeren. Dan is

$$S(B') = S(B) - 2^{-h_j} + 2^{-(h_j+1)} \cdot k \leq S(B) \leq 1 \ .$$

Met het inductieprincipe voor binaire bomen volgt dat $S(B) \leq 1$ voor alle B. In het geval van volledige binaire bomen kan hetzelfde bewijs worden geleverd, maar nu met $k = 2$, zodat gelijkheid geldt.

□

We mogen nu concluderen dat elke volledige binaire boom een kansverdeling bepaalt over zijn bladeren $(x_1, x_2, \ldots, x_n)$, namelijk

$$(q_1, q_2, \ldots, q_n) := (2^{-h_1}, 2^{-h_2}, \ldots, 2^{-h_n}) \ .$$

We noemen deze de *eigen kansverdeling* van de boom. Ten opzichte van deze kansverdeling is de gemiddelde hoogte van de bladeren gegeven door

$$\sum_{j=1}^{n} q_j h_j = -\sum_{j=1}^{n} q_j \log_2 q_j = H(\mathbf{q}) \ .$$

Dus de Shannon-entropie van de eigen kansverdeling van een boom is de gemiddelde bladhoogte bij die kansverdeling.

Naar aanleiding hiervan definiëren we de *reële entropie* van een kansverdeling $\mathbf{p}$ in een boom (vraagstrategie) als de gemiddelde bladhoogte (benodigde aantal vragen) van die boom met betrekking tot die verdeling:

$$H(B, \mathbf{p}) := \sum_{j=1}^{n} h_j^B p_j \ .$$

We hebben gezien dat voor de eigen kansverdeling $\mathbf{q}$ van B geldt:

$$H(B, \mathbf{q}^B) = H(\mathbf{q}^B) \ .$$

Lemma 3. *Voor elk tweetal kansverdelingen* $(p_1, p_2, \ldots, p_n)$ *en* $(q_1, q_2, \ldots, q_n)$ *geldt de ongelijkheid:*

$$-\sum p_i \log_2 q_i \geq -\sum p_i \log_2 p_i \ ,$$

waarbij we $0 \log 0 = 0$ *definiëren. Gelijkheid wordt alleen bereikt als* $p_i = q_i$ *voor alle* i*.*

Bewijs. Het is voldoende, de stelling te bewijzen voor de natuurlijke logaritme. We gaan uit van de bekende ongelijkheid voor een positief getal x:

$$\log x \leq x - 1 \ .$$

Hier volgt uit dat voor alle $p, q > 0$:

$$p(\log q - \log p) = p \log \frac{q}{p} \leq p \left(\frac{q}{p} - 1 \right) = q - p \ .$$

En dus, als $\sum p_i = \sum q_i = 1$:

$$\sum p_i \log q_i - \sum p_i \log p_i \leq \sum q_i - \sum p_i = 0 \ .$$

Gelijkheid geldt alleen voor $x = 1$ respectievelijk $p = q$, $p_i = q_i$. Door het nemen van limieten kan de ongelijkheid worden uitgebreid naar kansverdelingen waar nullen in voorkomen. □

Corollarium 4.

(i) Elke boom geeft voor zijn eigen kansverderling de optimale vraagstrategie:

$$\forall_B \forall_A : \quad H(A, \mathbf{q}^B) \geq H(B, \mathbf{q}^B) \quad (= H(\mathbf{q}^B)) \ .$$

(ii) Het verwachte aantal vragen bij kansverdeling $\mathbf{p}$ *is voor elke strategie minstens* $H(\mathbf{p})$*:*

$$\forall_B \forall_{\mathbf{p}} : \quad H(B, \mathbf{p}) \geq H(\mathbf{p}) \ .$$

Bewijs. (i): $\mathbf{q}$ is de eigen kansverdeling van een volledige binaire boom B, en zij A een andere binaire boom (vraagstrategie), waarin de bladeren, zeg, de hoogten $h_1^A, h_2^A, \ldots, h_n^A$ hebben.
Dan heeft A eigen kansverdeling $q_i^A = 2^{-h_i^A}$, en er geldt wegens Lemma 3:

$$\begin{aligned} H(A, \mathbf{q}^B) &= \sum q_i^B h_i^A = -\sum q_i^B \log_2 q_i^A \\ &\geq -\sum q_i^B \log_2 q_i^B \\ &= H(B, \mathbf{q}^B) = H(\mathbf{q}^B) \ . \end{aligned}$$

Dat wil zeggen: het gemiddelde aantal vragen bij strategie A is hoger dan dat bij B zelf.

(ii): Zij $\mathbf{p} = (p_1, p_2, \ldots, p_n)$ een willekeurige kansverdeling, en B een willekeurige volledige binaire boom met bladhoogten $(h_1, h_2, \ldots, h_n)$ en eigen kansverdeling $\mathbf{q} = (q_1, q_2, \ldots q_n)$. Dan geldt voor de reële entropie (=het gemiddelde aantal vragen):

$$H(B, \mathbf{p}) = \sum p_i h_i = -\sum p_i \log_2 q_i \geq -\sum p_i \log_2 p_i = H(\mathbf{p}) \ .$$

□

Hiermee is het eerste deel van Shannon's broncoderingsstelling (Stelling 1) bewezen.

3. Meerlettercoderingen

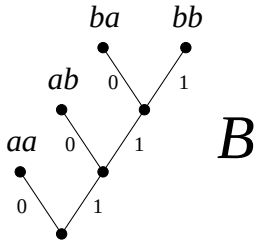
Voor het tweede deel van Stelling 1, dat het combineren van meerdere spelletjes (of codering van blokken van meer letters) betreft, bekijken we eerst weer een voorbeeld:

$$\mathbf{p} = (\tfrac{3}{4}, \tfrac{1}{4}) \ .$$

Het betreft hier een alfabet van twee letters: zeg a en b. De a komt driemaal zo vaak voor als de b, maar bij een codering die letter-voor-letter werkt kun je hier geen profijt van trekken: voor elke letter is één bit informatie nodig. We zeggen: de *reële entropie* van éénlettercodering is 1 bit. Echter als we de letters in de boodschap onafhankelijk veronderstellen, hebben de *paren* letters (aa, ab, ba, bb) de kansverdeling

$$\mathbf{p} \times \mathbf{p} = (\tfrac{9}{16}, \tfrac{3}{16}, \tfrac{3}{16}, \tfrac{1}{16}) \ .$$

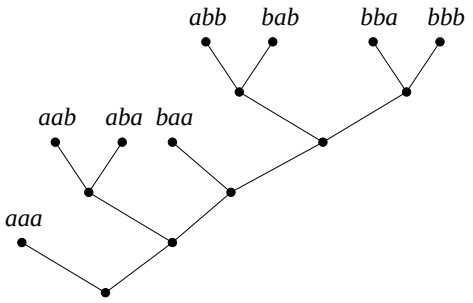
We kunnen daar het volgende boompje bij maken:



Dus aa coderen we met 0, ab met 10, ba met 110, en bb met 111. De reële entropie, het verwachte aantal bits bij gebruik van deze code, is de gemiddelde hoogte van de bladeren van het bijpassende boompje:

$$H(B, \mathbf{p} \times \mathbf{p}) = \tfrac{1}{16}(9 \cdot 1 + 3 \cdot 2 + 3 \cdot 3 + 1 \cdot 3) = \tfrac{27}{16} < 2 \text{ (bits)} .$$

Per letter is dit: $\frac{1}{2} \cdot \frac{27}{16} \approx 0{,}84375 \cdots$ bit.
Herhaling van de procedure voor blokken van drie letters levert de kansverdeling $\mathbf{p} \times \mathbf{p} \times \mathbf{p} = \frac{1}{64}(27, 9, 9, 9, 3, 3, 3, 1)$. We kunnen daarbij bijvoorbeeld deze boom ontwerpen. (In paragraaf 3 zullen we onthullen hoe we deze hebben gevonden.)



Nu komen we op een reële entropie van

$$\tfrac{1}{64}(27 + 3 \cdot 9 \cdot 3 + 3 \cdot 3 \cdot 5 + 5) = \tfrac{158}{64} = 3 \times 0{,}8229 \ldots$$

De bewering van Shannon is nu, dat de reële entropie per letter bij toenemende blokgrootte, en bij slim gekozen blokcodering, convergeert naar de entropie van $\mathbf{p}$:

$$H(\mathbf{p}) = -\tfrac{3}{4}\log_2 \tfrac{3}{4} - \tfrac{1}{4}\log_2 \tfrac{1}{4} = \log_2 4 - \tfrac{3}{4}\log_2 3 \approx 0{,}811278 \ldots$$

Dit zullen we nu gaan bewijzen. We beginnen met een omgekeerde van Lemma 2.

Lemma 5. *Als voor het rijtje natuurlijke getallen* $\mathbf{h} := (h_1, h_2, \ldots, h_n)$ *geldt dat* $\sum_{i=1}^{n} 2^{-h_i} \leq 1$, *dan bestaat er een (eventueel onvolledige) binaire boom met bladhoogten* h_1, h_2, $\ldots$, h_n.

Bewijs. Zij a_k het aantal keren dat het natuurlijke getal k in het rijtje $\mathbf{h}$ vóórkomt, en zij $k_1 \leq k_2 \leq \ldots \leq k_m$ een opsomming van de waarden van k waarvoor $a_k > 0$. Dan is, volgens de aanname,

$$\sum_{j=1}^{m} a_{k_j} 2^{-k_j} \leq 1 .$$

We construeren nu een boom met de juiste bladhoogten als volgt. Maak eerst een volledige binaire boom met 2^{k_1} bladeren op hoogte k_1. Een aantal a_k hiervan gebruiken we als bladeren in de te construeren boom. Dit is mogelijk omdat

$a_{k_1} \leq 2^{k_1}$. Bouw op de overige $2^{k_1} - a_{k_1}$ vertices op hoogte k_1, door herhaald ontkiemen, een verdere volledige binaire boom tot hoogte k_2. Het aantal bladeren op hoogte k_2 is dan $2^{k_2-k_1}(2^{k_1} - a_{k_1})$, hetgeen meer is dan a_{k_2}. Immers:

$$2^{-k_2}a_{k_2} + 2^{-k_1}a_{k_1} \leq 1 \ ,$$

$$\text{zodat} \quad 2^{k_2-k_1}(2^{k_1} - a_{k_1}) = 2^{k_2}(1 - 2^{-k_1}a_{k_1}) \geq a_{k_2} \ .$$

Reserveer een aantal a_{k_2} hieruit, en bouw met de rest verder tot niveau k_3. Opnieuw is

$$2^{k_3-k_2}\left(2^{k_2-k_1}(2^{k_1} - a_{k_1}) - a_{k_2}\right) \geq a_{k_3} \ .$$

Reserveer hier a_{k_3} bladeren, etcetera, totdat hoogte k_m bereikt wordt. Reserveer daar a_{k_m} bladeren, en snoei de rest af, zodat de boom mogelijk onvolledig wordt. De gezochte boom is klaar. □

Lemma 6. *Voor elke kansvector* $\mathbf{p} = (p_1, p_2, \ldots, p_n)$ *is er een, eventueel onvolledige, binaire boom* B *met* n *bladeren op hoogten* $(h_1, h_2, \ldots, h_n)$*, zó dat*

$$H(\mathbf{p}) \leq \sum_{i=1}^{n} p_i h_i \leq H(\mathbf{p}) + 1 \ .$$

Bewijs. Zij m_i de afronding naar boven van $-\log_2 p_i$. Dan is

$$\sum_{i=1}^{n} \frac{1}{2^{m_i}} \leq \sum_{i=1}^{n} p_i = 1 \ .$$

Volgens Lemma 5 is er dan een, eventueel onvolledige, binaire boom B met hoogten m_i, $i = 1, \ldots, n$. Voor deze boom geldt

$$H(B, \mathbf{p}) = \sum_{i=1}^{n} p_i m_i \leq \sum_{i=1}^{n} p_i(-\log_2 p_i + 1) = H(\mathbf{p}) + 1 \ .$$

□

Om het bewijs van Shannon's stelling af te kunnen ronden, moeten we weten wat de entropie is van producten van kansverdelingen, zodat we blokcodes met verschillende blokgrootten met elkaar kunnen vergelijken.
Als $\mathbf{p} = (p_1, p_2, \ldots, p_n)$ en $\mathbf{q} = (q_1, q_2, \ldots, q_m)$, dan verstaan we onder $\mathbf{p} \times \mathbf{q}$ de productverdeling $\mathbf{r}$ op het Cartesisch product $\{1, 2, \ldots, n\} \times \{1, 2, \ldots, m\}$, gegeven door

$$r_{(i,j)} := p_i q_j \ .$$

Lemma 7. *Voor twee eindige kansvectoren* $\mathbf{q}$ *en* $\mathbf{p}$ *geldt*

$$H(\mathbf{p} \times \mathbf{q}) = H(\mathbf{p}) + H(\mathbf{q}) \ .$$

Bewijs.

$$H(\mathbf{p}\times\mathbf{q}) = -\sum_{i=1}^{n}\sum_{j=1}^{m} p_i q_j \log_2(p_i q_j) = -\sum_{i=1}^{n}\sum_{j=1}^{m} p_i q_j (\log_2 p_i + \log_2 q_j)$$
$$= -\sum_{i=1}^{n} p_i \log_2 p_i - \sum_{j=1}^{m} q_j \log_2 q_j = H(\mathbf{p}) + H(\mathbf{q})\ .$$

□

Uit Lemma 7 volgt dat $H(\mathbf{p}^k) = kH(\mathbf{p})$.
Nu kunnen we het bewijs van Stelling 1afronden: Op grond van Lemma 6 is er voor elke $k \in \mathbb{N}$ een k-letter-code gegeven door een boom B_k zó dat

$$kH(\mathbf{p}) = H(\mathbf{p}^k) \le \sum_{i_1=1}^{n}\sum_{i_2=1}^{n}\cdots\sum_{i_k=1}^{n} p_{i_1}p_{i_2}\cdots p_{i_k} h^{B_k}_{i_1,i_2,\ldots,i_k} \le kH(\mathbf{p}) + 1\ .$$

Deling door k levert

$$H(\mathbf{p}) \le \frac{1}{k}H(\mathbf{p}^k, B_k) \le H(\mathbf{p}) + \frac{1}{k}\ .$$

Hierin is $\frac{1}{k}H(\mathbf{p}^k, B_k)$ de verwachting van het aantal bits per letter dat in de codering B_k nodig is om een blok van k letters over te seinen. □

4. Huffman-codering

De codering die in het bewijs van de Stelling van Shannon werd gevonden, is vaak niet optimaal. In 1952 heeft Huffman een eenvoudig algoritme gevonden dat bij een gegeven eindige discrete kansverdeling *de beste codering* oplevert. Construeer een binaire boom als volgt. Gegeven is een willekeurige kansvector $\mathbf{p} := (p_1, p_2, \ldots, p_n)$.

Algoritme van Huffman:
Teken eerst bij elk van de gewichten p_1, $p_2, \ldots, p_n$ een blad. Verbind vervolgens de bladeren met de kleinste twee gewichten met elkaar via een lager gelegen vertex, en hecht aan deze nieuw gecreëerde vertex het gezamenlijk gewicht van de twee bladeren. Herhaal dit tot alle bladeren tot één boom verbonden zijn.

Bij de 2-en 3-blokscoderingen in Paragraaf 2 hebben we dit algoritme al gebruikt. Dat zo de beste codering wordte gevonden, berust op de volgende stelling.

Stelling 8. *Zij $\mathbf{p} = (p_1, p_2, \ldots, p_n)$ een kansvector, zó geordend dat $p_1 \le p_2 \le \ldots \le p_n$. Zij B' een optimale boom voor $\mathbf{p}' := (p_1 + p_2, p_3, \ldots, p_n)$. Dan is B, die uit B' wordt verkregen door het blad met gewicht $p_1 + p_2$ te laten 'ontkiemen', een optimale boom voor $\mathbf{p}$.*

Bewijs. Zij A een willekeurige binaire boom met n bladeren. We willen aantonen dat $H(A, \mathbf{p}) \geq H(B, \mathbf{p})$.
Neem aan dat de nummers 1 en 2 naast elkaar boven in de boom A zitten. (Als dit niet zo is, breng ze dan door verwisselingen naar zulke posities toe. Dit zal de gemiddelde hoogte alleen kunnen verlagen.) Zij A' verkregen uit A door de bladeren 1 en 2 af te plukken, en hun gezamenlijke gewicht aan het stompje te hangen. Dan is

$$\begin{aligned} H(A', \mathbf{p}') &= (p_1 + p_2)(h_1 - 1) + \sum_{i=3}^{n} p_i h_i \\ &= -(p_1 + p_2) + \sum_{i=1}^{n} p_i h_i \\ &= H(A, \mathbf{p}) - (p_1 + p_2) \ . \end{aligned}$$

Dezelfde relatie geldt voor B en B'. Omdat B' optimaal is voor $\mathbf{p}'$, is $H(A', \mathbf{p}') \geq H(B', \mathbf{p}')$, dus geldt ook

$$H(A, \mathbf{p}) \geq H(A', \mathbf{p}') + (p_1 + p_2) \geq H(B', \mathbf{p}') + (p_1 + p_2) = H(B, \mathbf{p}) \ .$$

□

Nu volgt dat het algoritme van Huffman de beste codering oplevert:

Corollarium 9. *Zij $(p_1, p_2, \ldots, p_n)$ een kansvector, en zij B de volledige binaire boom die hieruit verkregen wordt met het Huffman-algoritme. Dan is de vraagstrategie beschreven door B optimaal voor de kansverdeling $\mathbf{p}$.*

Bewijs. Het Huffman-algoritme levert een rijtje bomen $B_0, B_1, B_2, \ldots, B_m$ en een rijtje verdelingen $\mathbf{p}_0, \mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_m$ op, met $B_0 = \{\odot\}$, $\mathbf{p}_0 = 1$, $B_m = B$, $\mathbf{p}_m = \mathbf{p}$ en zó dat steeds $(B_j, \mathbf{p}_j)$ uit $(B_{j+1}, \mathbf{p}_{j+1})$ wordt verkregen door de bladeren met de kleinste twee gewichten te fuseren. Het is duidelijk dat B_0 optimaal is voor de kansvector 1. Uit Stelling 8volgt nu inductief dat B_j optimaal is voor $\mathbf{p}_j$ voor $j = 1, 2, \ldots, m$, dus in het bijzonder dat B optimaal is voor $\mathbf{p}$. □

5. Entropie en Correlatie

Het begrip entropie kan, net als het in dit verband veel bekendere begrip correlatie, worden gebruikt om de afhankelijkheid tussen verschillende stochastische variabelen te quantificeren.
Laten X en Y discrete stochastische variabelen zijn op een kansruimte $(\Omega, \Sigma, \mathbb{P})$. Veronderstel dat X de waarden $(x_1, x_2, \ldots, x_n)$ kan aannemen met de kansen $(p_1, p_2, \ldots, p_n)$, en Y de waarden $(y_1, y_2, \ldots, y_m)$ met de kansen $(q_1, q_2, \ldots, q_m)$. Onder de *entropie* $H(X)$ van X verstaat men de entropie $H(\mathbf{p})$ van zijn kansverdeling. Onder onder de *gezamenlijke entropie* $H(X, Y)$ verstaan men de entropie van de gezamenlijke kansverdeling $(p_{i,j})$ van X en Y:

$$p_{i,j} := \mathbb{P}[X = x_i \quad \text{en} \quad Y = y_j] \qquad (i = 1, \ldots, n; \ \ j = 1, \ldots, m).$$

Propositie 10. *Voor discrete stochastische variabelen X en Y geldt*

$$H(X) \leq H(X,Y) \leq H(X) + H(Y) \; .$$

De tweede ongelijkheid is verzadigd desda X en Y onafhankelijk zijn; de eerste desda er een functie $\varphi : (x_1, x_2, \ldots, x_n) \to (y_1, y_2, \ldots, y_m)$ bestaat zó dat met kans 1 geldt:

$$Y = \varphi(X) \; .$$

Bewijs. Voor de rechter ongelijkheid breiden we het bewijs van Lemma 7 iets uit. Op grond van Lemma 3 geldt:

$$\begin{aligned} H(X) + H(Y) &= -\sum_{i=1}^{n}\sum_{j=1}^{m} p_{i,j}(\log_2 p_i + \log_2 q_j) \\ &= -\sum_{i=1}^{n}\sum_{j=1}^{m} p_{i,j}\log_2 p_i q_j \geq -\sum_{i=1}^{n}\sum_{j=1}^{m} p_{i,j}\log_2 p_{i,j} = H(X,Y) \; . \end{aligned}$$

Gelijkheid geldt desda $p_{i,j} = p_i q_j$ voor alle i en j, dat wil zeggen als $X \perp\!\!\!\perp Y$. Voor de linker ongelijkheid voeren we in:

$$a_{i,j} := \mathbb{P}[Y = y_j | X = x_i] \; ,$$

zodat geldt: $p_{i,j} = p_i a_{i,j}$. We vinden:

$$\begin{aligned} H(X,Y) - H(X) &= -\sum_{i=1}^{n}\sum_{j=1}^{m} p_{i,j}(\log_2 p_{i,j} - \log_2 p_i) \\ &= -\sum_{i=1}^{n}\sum_{j=1}^{m} p_{i,j}\log_2 a_{i,j} \\ &= -\sum_{i=1}^{n} p_i \sum_{j=1}^{m} a_{i,j}\log_2 a_{i,j} \geq 0. \end{aligned}$$

Gelijkheid geldt desda voor die waarden van i waarvoor $p_i > 0$, de kansverdeling $j \mapsto a_{i,j}$ entropie 0 heeft; dat wil zeggen als alle $a_{i,j}$ gelijk zijn aan 0 of 1. Omdat voor alle i (met $p_i > 0$) geldt dat $\sum_{j=1}^{m} a_{i,j} = 1$, is er precies één j zó dat $a_{i,j} = 1$. Definieer nu $\varphi(x_i) := y_j$, (en voor het gemak ook $\varphi'(i) := j$), en er volgt

$$\mathbb{P}[Y = \varphi(X)] = \sum_{i=1}^{n} \mathbb{P}[Y = \varphi(x_i) | X = x_i] \cdot \mathbb{P}[X = x_i] = \sum_{i=1}^{n} a_{i,\varphi'(i)} p_i = \sum_{i=1}^{n} p_i = 1 \; .$$

□

Deze ongelijkheden zijn aanleiding tot het definiëren van twee nieuwe begrippen:

DEFINITIE. Onder de *wederzijdse entropie* $I(X,Y)$ van twee stochasten X en Y verstaan we het verschil

$$H(X) + H(Y) - H(X,Y) \; .$$

Uit de propositie volgt direct dat deze grootheid positief is, en dat $I(X,Y) = 0$ desda X en Y onafhankelijk zijn. Ook is duidelijk dat hij symmetrisch is in X en Y. We kunnen deze wederzijdse entropie interpreteren als de hoeveelheid informatie die X bevat *over* Y, en omgekeerd. In het ene extreme geval ($I(X,Y) = 0$) onthult de waarde van X niets over Y, en omgekeerd. In het andere extreme geval ($I(X,Y) = H(Y)$, oftewel $H(X,Y) = H(X)$), is Y door X geheel vastgelegd.

DEFINITIE. Onder de *voorwaardelijke entropie* $H(Y|X)$ van Y, gegeven X, verstaan we het verschil

$$H(X,Y) - H(X) = \sum_{i=1}^{n} \mathbb{P}[X = x_i] \sum_{j=1}^{m} \mathbb{P}[Y = y_j | X = x_i] \log_2 \mathbb{P}[Y = y_j | X = x_i] \; .$$

(De uitdrukking rechts wordt in het bewijs van Propositie 10 verklaard.) De voorwaardelijke entropie kan worden gezien als de hoeveelheid informatie die een onthulling van Y nog kan opleveren als X al bekend is. In het ene extreme geval ($H(Y|X) = H(Y)$) is alle informatie over Y nog een verrassing, omdat X en Y onafhankelijk zijn, zodat we over Y nog niets weten. In het andere extreme geval ($H(Y|X) = 0$) is $Y = \varphi(X)$, dus al volledig bekend.

5.1 *Covariantie*

Het is illustratief, bovenstaande begrippen te vergelijken met de bekendere begrippen *covariantie* en *correlatie*.

Propositie 11. *Laten X en Y discrete stochasten zijn, en veronderstel voor het gemak dat* $\mathrm{Var}\,(\mathrm{X}) > 0$. *Er geldt*

$$0 \leq \mathrm{Cov}\,(\mathrm{X},\mathrm{Y})^2 \leq \mathrm{Var}\,(\mathrm{X})\mathrm{Var}\,(\mathrm{Y}) \; ,$$

Gelijkheid links geldt als X en Y onafhankelijk zijn, maar het omgekeerde is niet het geval. Gelijkheid rechts treedt op alleen als $Y = \varphi(X)$ voor een lineaire functie $\varphi : \mathbb{R} \to \mathbb{R}$.

Bewijs. De linkerongelijkheid is triviaal. Gelijkheid treedt op als X en Y ongecorreleerd zijn, wat zoals bekend zwakker is dan onafhankelijk. Voor het bewijs van de rechter-ongelijkheid merken we op dat voor alle $\lambda \in \mathbb{R}$ geldt:

$$0 \leq \mathrm{Var}\,(\mathrm{Y} - \lambda \mathrm{X}) = \lambda^2 \mathrm{Var}\,(\mathrm{X}) - 2\lambda \mathrm{Cov}\,(\mathrm{X},\mathrm{Y}) + \mathrm{Var}\,(\mathrm{Y}) \; .$$

De discriminant van de kwadratische functie in het rechterlid moet dus ≤ 0 zijn. Dit is de rechterongelijkheid. Hier treedt gelijkheid op desda $\mathrm{Var}\,(\mathrm{Y} - \lambda \mathrm{X}) = 0$, dat wil zeggen als

$$Y = \lambda X + \mu$$

voor zekere constante μ. □

Het belangrijkste verschil tussen covariantie en wederzijde entropie, gezien als maat voor de samenhang van twee grootheden X en Y, is dat de covariantie kwadratisch is in de waarden x_i en y_j, terwijl het voor de bepaling van de entropie alleen nodig is, deze waarden van elkaar te kunnen onderscheiden. Covariantie kijkt naar X en Y als elementen van $L^2(\Omega, \Sigma, \mathbb{P})$, wederzijdse entropie alleen naar de door hen voortgebrachte partities van Ω.

5.2 *Een toepassing: beeldregistratie.*

In de medische literatuur vind je een toepassing van wederzijds entropie op het probleem van *beeldregistratie*: het correct over elkaar heen leggen van beelden, gemaakt met verschillende opname-apparaten, zoals scans met behulp van MRI ("Magnetic Resonance Imaging") enerzijds en PET ("Positron Emission Tomography") anderzijds.

Hierbij worden twee- of driedimensionale beelden, te beschouwen als afbeeldingen X en Y van delen A en B van $\mathbb{Z}^2$ of $\mathbb{Z}^3$ naar twee paletten $(x_1, x_2, \ldots, x_n)$ en $(y_1, y_2, \ldots, y_m)$, in verschillende standen met elkaar vergeleken. In elke stand $\psi : A \to B$ wordt de entropie berekend van de empirische gezamenlijke verdeling van X en $Y \circ \psi$:

$$p_{i,j} := \frac{\#\{\, a \in A \mid X(a) = i, Y(\psi(a)) = j \,\}}{\#(A)} \,.$$

De afbeelding ψ (meestal een combinatie van een translatie, een rotatie en een schaalfactor) waarin het beeld X de meeste informatie bevat over het beeld $Y \circ \psi$ blijkt goed overeen te komen met de inhoudelijke correspondentie tussen de twee beelden.

Bij het probleem van beeldregistratie zou men ook de covariantie van de gezamenlijke kansverdeling kunnen gebruiken. Deze is in het algemeen gepronnonceerder dan de gezamenlijke entropie, mits de grootheden X en Y de neiging hebben, steeds dezelfde kant uit te varieëren bij een overgang van het ene soort weefsel naar het andere. Is zo'n verband er niet, dan zal de entropie het beter doen.

5.3 *Normering.*

De *genormeerde wederzijdse informatie* (*"Normalized Mutual Information"*) wordt gegeven door

$$NMI(X, Y) := \frac{H(X) + H(Y)}{H(X, Y)} \,;$$

een variant hierop is de *entropie-correlatiecoëfficient*

$$ECC(X,Y) := \frac{I(X,Y)}{H(X)+H(Y)} = 1 - \frac{H(X,Y)}{H(X)+H(Y)} \ .$$

Voor de genormeerde wederzijdse informatie corresponderen de twee extreme gevallen met de waarden 1 en $1 + H(Y)/H(X)$.

Ook de covariantie heeft zijn genormeerde versie: de correlatiecoëfficient:

$$\rho_{X,Y} := \frac{\mathrm{Cov\,(X,Y)}}{\sqrt{\mathrm{Var\,(X)Var\,(Y)}}} \ ,$$

die waarden aanneemt, variërend van -1 tot 1.

Deze genormeerde varianten van gezamenlijke entropie en covariantie hebben voor de hierboven genoemde toepassing het voordeel dat zij minder gevoelig zijn voor ongewenste randeffecten. Een van die effecten is bijvoorbeeld dat de covariantie van X en Y groter wordt wanneer de twee te vergelijken beelden een grotere overlap vertonen.

De tweede pijler onder de informatietheorie is de tweede stelling van Shannon, het *Noisy Coding Theorem*. Deze stelling is aanzienlijk lastiger te bewijzen dan de ruisloze coderingsstelling, en we zullen ons in dit bestek tevreden moeten stellen met een formulering van de stelling en een vage indicatie waarom hij waar is.

Ook in de tweede stelling van Shannon spelen codering en Shannon's informatiebegrip een sleutelrol, maar de vraagstelling is een andere. In de eerste stelling ging het erom, de hoeveelheid informatie in een boodschap te karakteriseren. Deze werd vastgesteld als de minimale compressiefactor, die gegeven wordt door de Shannon-informatie.

De tweede stelling beantwoordt de vraag, hoeveel informatie er per tijdseenheid door een 'kanaal' kan worden getransporteerd. In de beginjaren van de informatietheorie stelde dit kanaal een transmissielijn of een radiozender voor, tegenwoordig worden er ook processoren en DVD-sporen mee gemodelleerd. Het wiskundige model is steeds hetzelfde gebleven.
Het kanaal wordt abstract voorgesteld als een kastje, een *'black box'*, met een ingang en een uitgang, en een tikkende klok. Aan de ingang kan op elke tik van de klok een symbool X worden ingevoerd uit een alfabet $\mathcal{X} := (x_1, x_2, \ldots, x_n)$, en aan de uitgang wordt een symbool Y afgelezen uit een alfabet $\mathcal{Y} := (y_1, y_2, \ldots, y_m)$ dat stochastisch is en alleen afhangt van de input X. Het kastje wordt gekarakteriseerd door een overgangsmatrix A met ingangen

$$a_{ij} := \mathbb{P}[Y = y_j | X = x_i] \ .$$

Dit kastje met zijn klok wordt het *discrete geheugenloze kanaal* genoemd.
Stel je voor dat je over zo'n kanaal een boodschap wilt versturen.
Als uit de output Y éénduidig kan worden opgemaakt wat de input X geweest is, dan levert het kanaal eigenlijk geen nieuw probleem op: de hoeveelheid informatie die per tijdseenheid kan worden verstuurd is $\log n$. Bijvoorbeeld: als $n = 2$, $m = 4$, en A heeft de gedaante

$$A = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix} \ ,$$

dan kan er elke seconde aan de output worden afgelezen, wat de input was, en is het informatietransport 1 bit/seconde. In dit voorbeeld geldt dat de output $\mathcal{Y}$ kan worden verdeeld in disjuncte blokken $\mathcal{Y}_i$, $i = 1, \ldots n$, zó dat door een input x_i alleen een symbool in $\mathcal{Y}_i$ geproduceerd kan worden. Merk op dat dit equivalent is met

$$H(Y|X) = 0 \quad \text{voor alle kansverdelingen } \mathbf{p} \text{ van } X \ .$$

De capaciteit van het kanaal is gelijk aan de entropie van de bron.

Anders wordt het, als verschillende inputs tot identieke output kunnen leiden. Het eenvoudigste voorbeeld is het zogenaamde binaire symmetrische kanaal, dat gegeven wordt door de overgangsmatrix

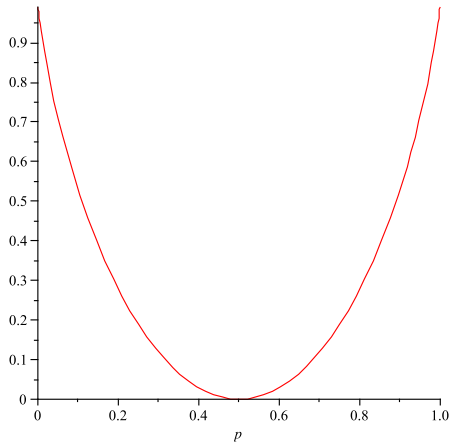
$$A = \begin{pmatrix} 1-p & p \\ p & 1-p \end{pmatrix} .$$

Het inputbit X wordt met kans $1-p$ getrouw doorgestuurd naar de output, maar met kans p omgeklapt. Laten we eens uitrekenen hoeveel informatie de output Y bevat over de input X.

$$\begin{aligned} I(X,Y) &= H(X) + H(Y) - H(X,Y) = H(Y) - H(Y|X) \\ &= H(Y) - \sum_{i=1}^{n} p_i H(Y|X = x_i) = H(Y) - H(p, 1-p) \leq 1 - H(p, 1-p) . \end{aligned}$$

De wederzijds informatie hangt dus nog van $H(Y)$ af, maar omdat we geinteresseerd zijn in maximale informatie-overdracht, kiezen we de kansverdeling van X zó, dat $H(Y)$ zijn maximale waarde 1 bereikt. Dit is het geval voor de gelijkverdeling $(\frac{1}{2}, \frac{1}{2})$ over de inputbits (en dan ook over de outputbits). Dit maximum over de input-kansvedeling van de informatie die de output bevat *over* de input, noemt men de *capaciteit* C van het kanaal. Voor het binaire symmetrische kanaal met parameter p is de capaciteit dus

$$C = 1 - H(p, 1-p) = 1 + p\log_2 p + (1-p)\log_2 (1-p) .$$



Capaciteit van binair symmetrisch kanaal

DEFINITIE. Onder de *informatiecapaciteit* C van een discreet geheugenloos kanaal verstaat men

$$C := \max_{\mathbf{p}} I(X,Y) \ ,$$

waarbij het maximum wordt genomen over de kansverdeling $\mathbf{p}$ van de input X.
De grote ontdekking van Shannon is de volgende. (Onze formulering is informeel.)

Stelling 12. *Voor elke gewenste informatie-'rate'* $R < C$*, en voor willekeurig kleine* $\varepsilon > 0$ *is het mogelijk, een blok-code te construeren, zó dat per tijdseenheid een hoeveelheid informatie* R *kan worden verstuurd met een kans op een fout kleiner dan* ε*.*

Dit resultaat is verbluffend, en werd ook in de jaren '40 (vóór Shannon) voor onmogelijk gehouden. Stel je een binair symmetrisch kanaal voor met bijvoorbeeld $p = \frac{1}{10}$. Van elke tien bits die je erin stuurt wordt er één omgeklapt. Je weet niet welke. Geen enkel bit is betrouwbaar. Maar $C = 1 + p\log_2 p + (1-p)\log_2(1-p) \approx 0{,}531\ldots$. Shannon beweert nu dat er een n-blok-code moet bestaan voor zekere $n \in \mathbb{N}$, waarin een willekeurige boodschap van n bits kan worden gecodeerd in $2n$ bits, over het kanaal verzonden en gedecodeerd, zó dat de kans op een fout kleiner is dan, zeg, één op duizend. Of één op miljard; met een wat grotere n is ook dat mogelijk.
De tweede stelling van Shanon voorspelt dus het bestaan van foutencorrigerende codes ("error correcting codes") van willekeurig grote betrouwbaarheid, mits de *rate* waarmee de oorspronkelijke boodschap wordt aangeboden kleiner blijft dan de capaciteit van het kanaal.
Het resultaat is gebaseerd op het volgende feit uit de kansrekening, bekend als *asymptotische equipartitie*: Zij $X_1, X_2, \ldots, X_n$ een rijtje onafhankelijke discrete stochasten met verdeling $\mathbf{p} = (p_1, \ldots, p_k)$. Dit definieert een kansverdeling op $\mathcal{X}^n$. De bewering is nu, dat voor grote n het grootste deel van de kans wordt opgeslokt door een deelverzameling van slechts $2^{nH(X)}$ rijtjes, elk met kans ongeveer $2^{-nH(X)}$. (Hierbij is $H(X)$ per definitie de entropie $H(\mathbf{p})$ van de verdeling van X.) Als gevolg hiervan zijn er enerzijds eigenlijk maar $2^{nH(Y)}$ outputrijtjes die werkelijk optreden, terwijl er anderzijds $2^{nH(Y|X)}$ rijtjes zijn bij elke input $x := (x_1, \ldots, x_n) \in \mathcal{X}^n$. De kunst is nu, een aantal van M codewoorden uit $\mathcal{X}^n$ te selecteren, zo dat de bijbehorende 'typische' outputverzamelingen $\mathcal{Y}_{x_\mu}$, $\mu = 1, \ldots, M$, allemaal vrijwel disjunct zijn. Dit kan alleen als

$$M < 2^{nH(Y)}/2^{nH(Y|X)} = 2^{n(H(Y)-H(Y|X))} = 2^{nI(X,Y)} \ .$$

Dàt het ook werkelijk kan, werd door Shannon beargumenteerd in 1948, en bewezen door Feinstein in 1952, en op een eenvoudiger manier door Gallager in 1965. Deze bewijzen zijn niet constructief: zij geven geen aanwijzing hoe de foutverbeterende codes kunnen worden gevonden.

6.1 *Foutverbeterende codes*

Deze situatie heeft aanleiding gegeven tot een enorme literatuur over foutverbeterende codes, van de allereenvoudigste repetitieve codes (zend elk bit k maal), via parity-check en Hamming-codes tot de moderne 'turbo-codes' die gebruikt worden in de ruimtevaart, en die zeer dicht in de buurt komen van de Shannon-limiet uit Stelling 12.

De bedoeling van zulke codes is, redundantie te introduceren, zodat het, ook als een deel van de informatie verloren gaat of vervormd wordt, toch nog mogelijk is om bij de ontvanger de oorspronkelijke boodschap terug te vinden. Het meest voor de hand liggende coderingsschema is herhaling: om een 1 te versturen, zenden we 11111 over het kanaal, en om een 0 te versturen, zenden we 00000. Zo hebben we vijf symbolen nodig voor één enkel bit, dus we krijgen een *rate* van $\frac{1}{5}$ bit per symbool. Als we deze code gebruiken voor het binaire symmetrische kanaal, dan is de optimale decodering de regel 'de meeste stemmen gelden' binnen elk blok van vijf ontvangen bits. We ontvangen dan alleen een foute boodschap als drie of meer bits van het blok zijn omgeslagen. Door langere blokken te gebruiken, kunnen we de foutkans willekeurig klein krijgen, maar de *rate* gaat bij toenemende bloklengte naar 0. Deze code is dus wel eenvoudig te implementeren, maar niet zo nuttig.

6.2 *Foutdetecterende codes*

Inplaats van de bits simpelweg te herhalen, kunnen we ze op slimmere wijze combineren, zó dat de extra bits controleren of er fouten in de andere zijn geslopen. Het eenvoudigste voorbeeld is pariteitscontrole: beginnend met een blok van $n-1$ informatie-bits, kiezen we het n-de bit zó dat de pariteit van het hele blok nul wordt (het aantal enen in het blok is even). Als we dan aan de ontvangerskant een oneven aantal enen binnenkrijgen, weten we dat er iets fout is gegaan. Dit is het eenvoudigste voorbeeld van een *foutendetecterende* code. Maar deze code detecteert een even aantal fouten niet, en geeft bovendien geen enkele aanwijzing hoe de fout moet worden hersteld.

6.3 *Hamming codes*

Het idee van pariteitscontrole kan worden uitgebreid door meer pariteitsbits te gebruiken, die de pariteit van verschillende delen van het blok controleren. De *Hamming-codes* vormen hier een mooi voorbeeld van. Dit zijn *lineaire codes*; zij maken gebruik maken van lineaire algebra over het lichaam $\mathbb{F}_2 = \{0, 1\}$ van de gehele getallen modulo 2.

Kies een natuurlijk getal n, en zij $2^{n-1} \leq m < 2^n$. (Het zuinigst is $m = 2^n - 1$.) Definieer nu de $n \times m$-matrix H door:

$$h_{ij} := \text{ het } i\text{-de bit in de binaire uitdrukking van } j \ .$$

Voor $n = 3$ en $m = 2^3 - 1 = 7$ vinden we zo

$$H = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} \ .$$

Beschouw nu de nulruimte $\mathcal{N}(H)$ van H, een lineaire deelruimte van $\mathbb{F}_2^m$ van dimensie $k := m - n$. Deze bestaat uit 2^k verschillende codewoorden ter lengte m. Eén hiervan is het woord $000\ldots0$, de nulvector in $\mathbb{F}_2^m$. Het aardige is nu, dat alle andere codewoorden minstens drie enen bevatten. Immers, als v één 1 bevat, zeg $v = e_i$, dan is Hv de i-de kolom van H, dus $v \notin \mathcal{N}(H)$. En als $v = e_i + e_j (= e_i - e_j)$ met $i \neq j$, dan is

$$Hv = H(e_i - e_j) = He_i - He_j \neq 0\ ,$$

want de kolommen van H zijn verschillend.
Als nu c_1 en c_2 verschillende codewoorden zijn, dat wil zeggen: vectoren in de nulruite van H, dan is ook $c_1 - c_2 \notin \mathcal{N}(H)$. Dan bevat $c_1 - c_2$ minstens drie enen, en dus zijn c_1 en c_2 op minstens drie plaatsen verschillend. Men zegt dat de *Hamming-afstand* tussen de codewoorden minstens 3 is; de codewoorden liggen mooi gespreid in de beschikbare code-ruimte $\mathcal{N}(H)$
Dit leidt tot de volgende goede eigenschappen van Hamming-codes:

(ii) De code kan in één woord twee fouten detecteren;

(i) De code kan in één woord één fout corrigeren.

- *Ad (i):* Foutendetectie: Pas H toe op de output, en accepteer deze alleen als $Hu = 0$. Omdat de Hamming-afstand tussen codewoorden minstens drie is, kan pas met drie fouten in de transmissie een verkeerd codewoord worden ontvangen.
- *Ad (ii):* Foutencorrectie: Pas H toe op de output u. Als de output u gelijk is aan de input c, dan is $Hu = 0$. In dat geval accepteren we de output zonder meer. Maar als er op één plaats i een fout is ingeslopen, dan zien we

  $$Hu = H(c + e_i) = Hc + He_i = He_i\ ,$$

  de vector in de i-de kolom van H. Maar dit is precies het getal i, geschreven in binaire notatie! We klappen het op deze manier verraden bit weer terug, en de fout is gecorrigeerd.
  Pas als er meer dan één fout wordt gemaakt bij de verzending van het codewoord, zal bovenstaand algoritme een verkeerd woord opleveren.

De foutencorrigerende Hamming-code heeft de volgende karakteristieken: de transmissierate is:

$$R = \frac{k}{m} = \frac{m-n}{m} = 1 - \frac{n}{m} \leq 1 - \frac{n}{2^n - 1}\ .$$

En de kans op een foute decodering van één blok is

$$\sum_{j=2}^{m} \binom{m}{j} p^j (1-p)^{m-j} = \binom{m}{2} p^2 + o(p)\ .$$

Merk op dat voor $n \to \infty$ de rate R en de foutenkans beide naar 1 gaan. Fraai als hij is, geeft deze code ons dus geen middel in de hand om de grens te benaderen, die door Shannon's tweede stelling (Stelling 12) wordt gegeven.