2 Informationstheorie

Formale Grundlagen der Informatik I

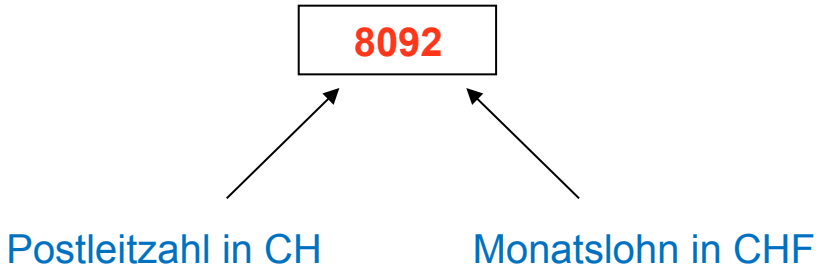
Herbstsemester 2012

Robert Marti

Vorlesung teilweise basierend auf Unterlagen
von Prof. emer. Helmut Schauer

Grundbegriffe

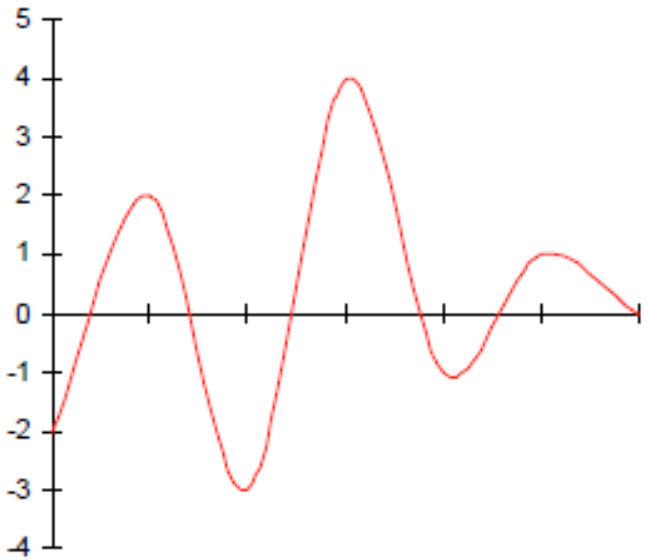
- Informatik
  (IT: Information Technology, ICT: Information & Communication Technology)
  - Disziplin / Wissenschaft der Informationsverarbeitung (Info. Processing)
- Signal
  - **analog**
  - **digital**
- Nachricht (message) – auch Daten (data)
  - Begriff Nachricht wird meist bei Datenübermittlung verwendet
- Information (Text, Bilder, Ton, …)
  - letztlich Nachrichten (Daten) + Kontext



Analoges Signal

- Beispiele:
  Radio, TV, Telefon, Magnetband, Vinyl Schallplatten, ...

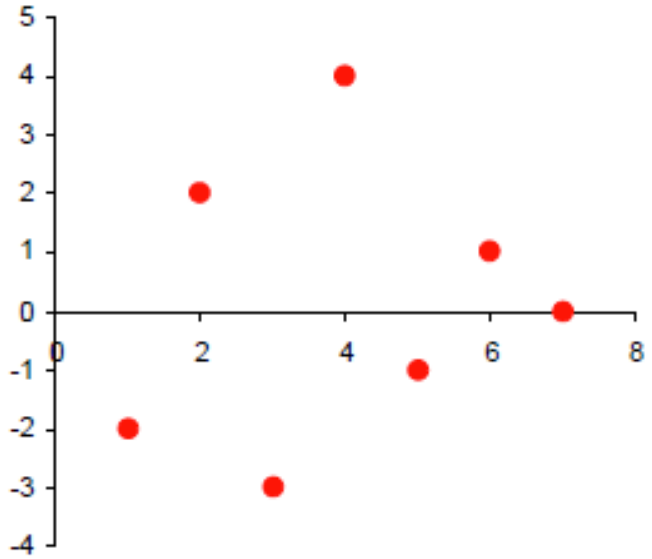
- Eigenschaften
  - kontinuierlich
  - im Prinzip beliebig genau
  - Verarbeitung eher aufwendig



Digitales Signal

- Beispiele:
  PC Spiele, VoIP (**V**oice **o**ver **I**nternet **P**rotocol), Digitalkameras, MP3 Musik (auch FLAC Musik), …

- Eigenschaften
  - diskrete Zustände
  - begrenzt genau (endlich viele verschiedene Zustände)
  - Verarbeitung weniger aufwendig



Nachricht

- Nachrichten (Daten) besitzen einen bestimmtes "Format", sie folgen einer (formalen) **Sprache**.

- Eine Sprache wird definiert durch

  – eine **Syntax** (bei natürlichen Sprachen Grammatik genannt)
  ➔ Repräsentation

  Beispiele (hier externe Repräsentationen in Textform):
  1. Feb. 2011, 01.02.11, 2/1/11, 2011-02-01, 32. Tag im 2011

  – eine **Semantik**
  ➔ Bedeutung

  Beispiel: Ziffern vor Punkt = Tag im Monat, danach folgen 3 Buchstaben für den Monat, danach 4 Ziffern für das Jahr

Bespiele für Sprachen

- Chemische Formeln

  H_2SO_4

- Beschreibung von Schachzügen

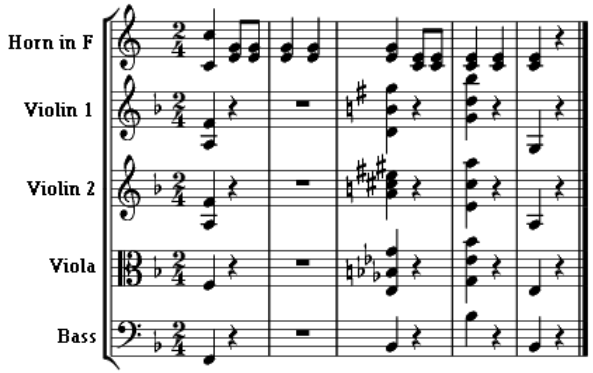
  b2 x c4!

- Mathematische Formelsprache

  (– b + sqrt(b*b – 4*a*c)) / (2*a) bzw.

$$\frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

- Musiknoten



- Programmiersprachen

```
s = 0; for (int i = 1; i <= n; i++)  s = s+i;
```

Syntax und Alphabet

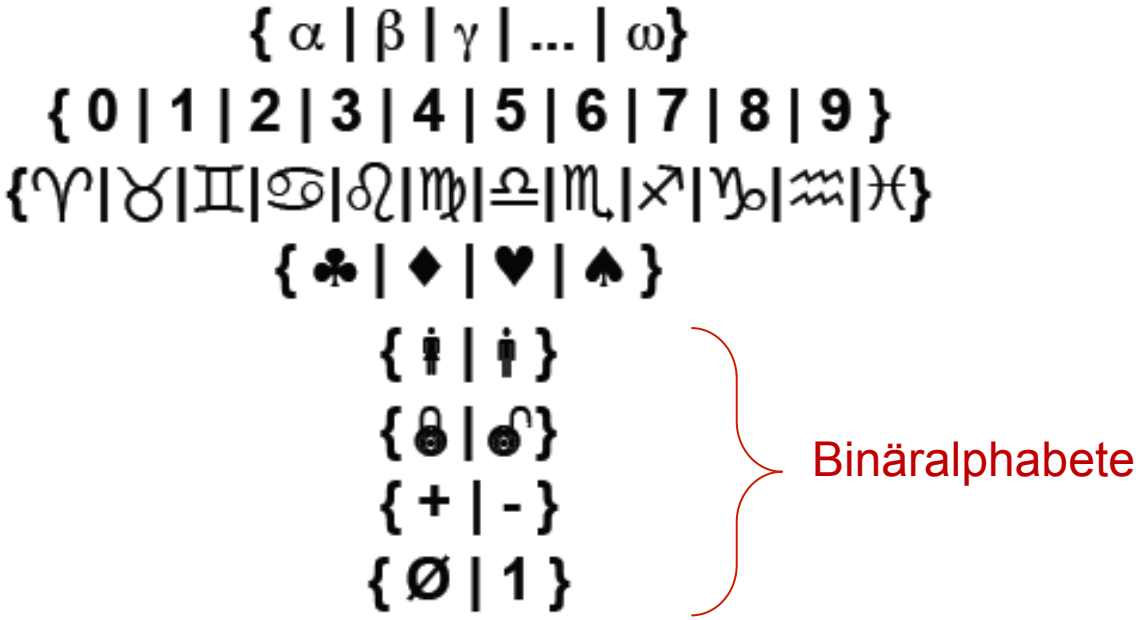
- Die Definition einer Syntax stützt sich auf ein **Alphabet**:
  Menge der zulässigen Zeichen

- Beispiele

  – 26 (Gross-) Buchstaben

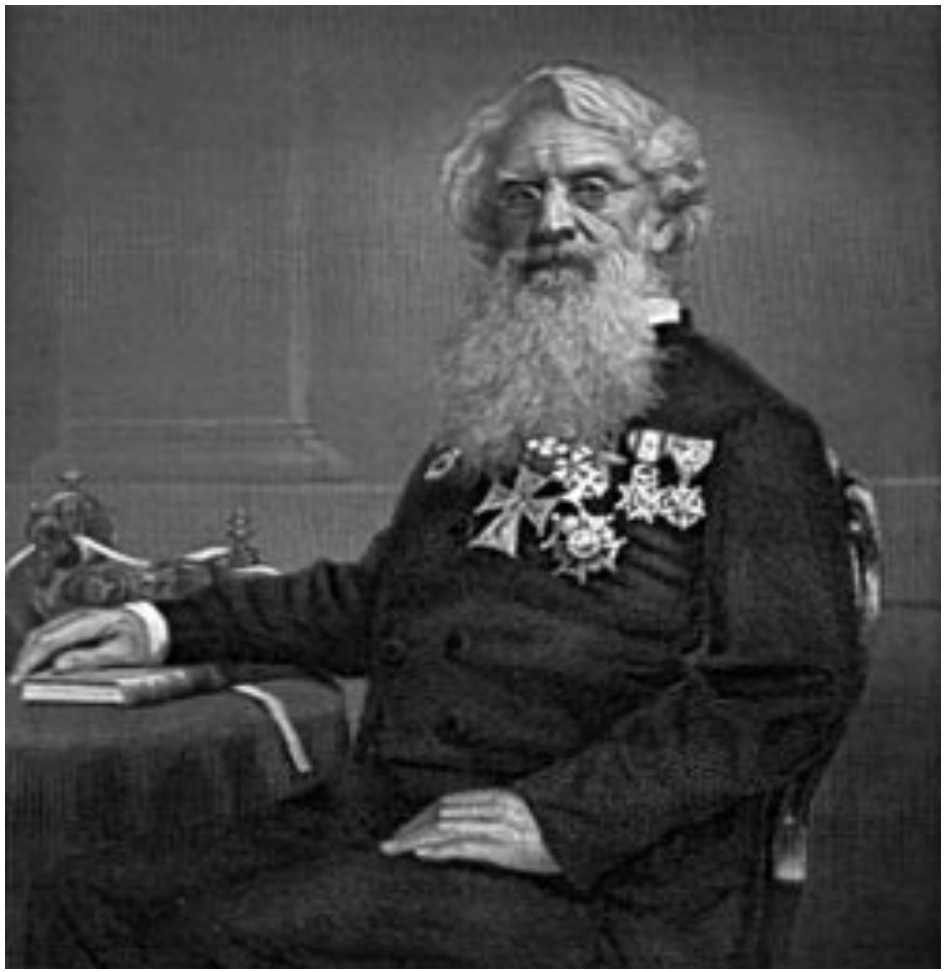
  – 2 x 26 Buchstaben, 10 Ziffern, Spezialzeichen, insges. ca. 100 Zeichen

  – andere:



Code

- Beispiel **Morsecode**: Code eines Alphabets mit Zeichen - und —



A	·–	N	–·		
B	–···	O	–––	1	·————
C	–·–·	P	·––·	2	··–––
D	–··	Q	––·–	3	···––
E	·	R	·–·	4	····–
F	··–·	S	···	5	·····
G	––·	T	–	6	–····
H	····	U	··–	7	––···
I	··	V	···–	8	–––··
J	·–––	W	·––	9	––––·
K	–·–	X	–··–	0	–––––
L	·–··	Y	–·––		
M	––	Z	––··		

Binärcode

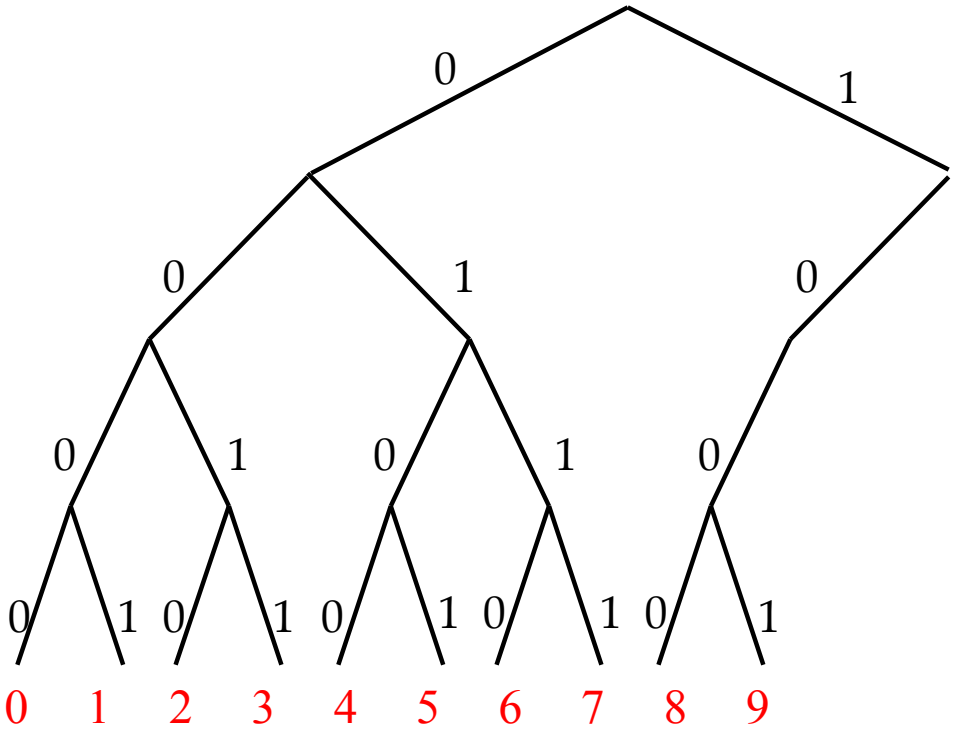
- interne Repräsentation in Digitalcomputern

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

- Bem.: Zahl 10 wird intern als 1010_2 (binär) oder evt. als $0001_2\ 0000_2$ (BCD = Binary Coded Decimal) dargestellt (z.B. Taschenrechner, Sprache COBOL)

Codebaum

- "Tree Walk" von Wurzel aus
  - Abstieg "nach links" ergibt binäre Ziffer 0
  - Abstieg "nach links" ergibt binäre Ziffer 1



Darstellung von Werten durch Bitfolgen

- Eine Bitfolge der Länge l (auch **Wortlänge** genannt) gestattet die Darstellung von n Werten, wobei:

$$n = 2^l$$

- Wenn n verschiedene (gleich wahrscheinliche) Werte dargestellt werden sollen, dann werden demnach im Minimum

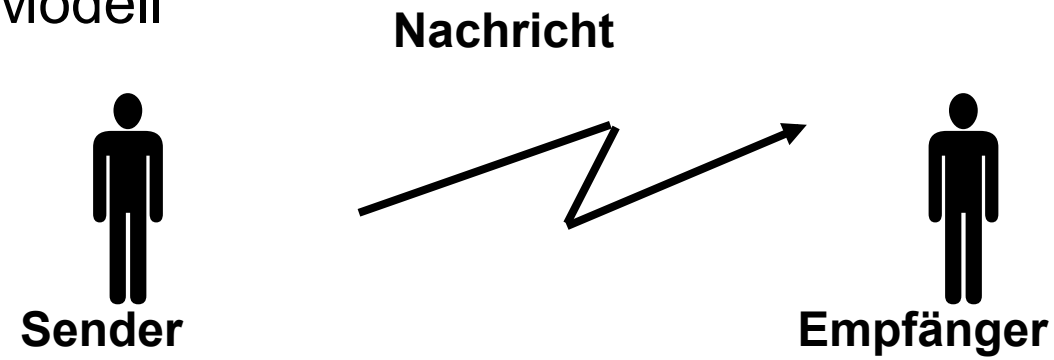
$\log_2 n$ (auch $\mathrm{ld}\, n$, wobei ld = **L**ogarithmus **D**ualis)

Bits benötigt

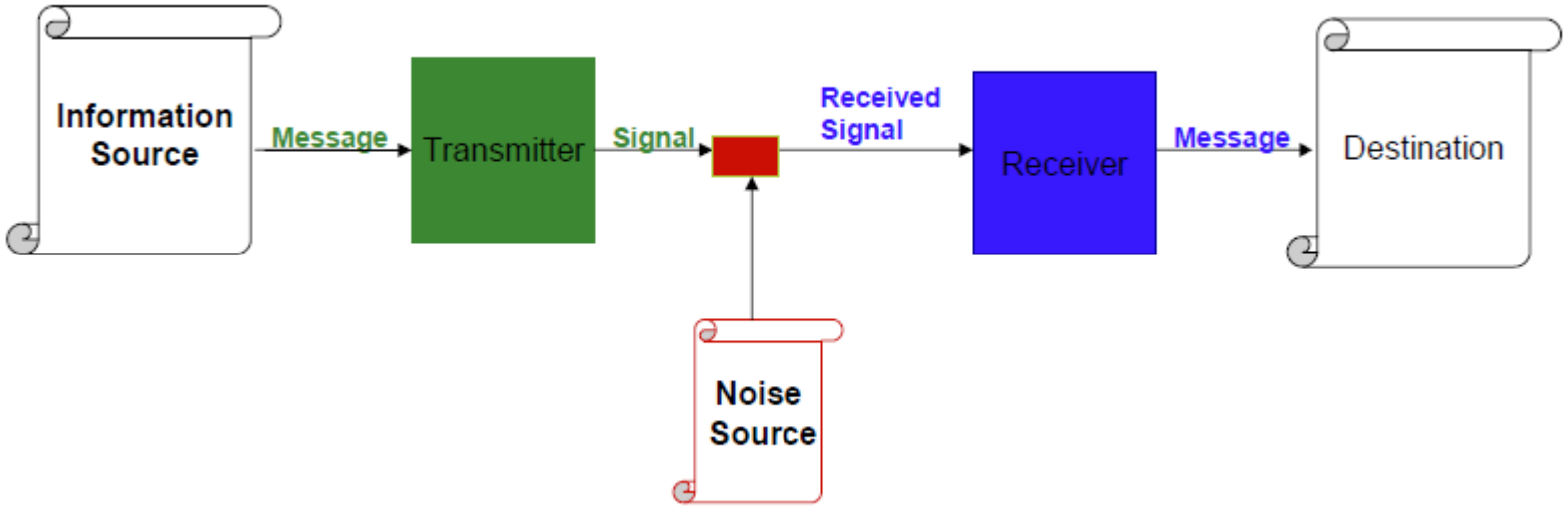
- Beispiele: Ganze Zahlen mit und ohne Vorzeichen
  unsigned integer: $[0, 2^l - 1]$, l meistens = 8, 16, 32, oder 64
  signed integer: $[-2^{l-1}, +2^{l-1} - 1]$ (1 Bit für Vorzeichen)

Ein abstraktes Modell für Kommunikation

- einfaches Modell



- etwas detaillierteres Modell



Messen des Informationsgehalts

- Grundidee:
  Wir ordnen einem Zufallsereignis, welches 2 mögliche, gleich wahrscheinliche Resultate hat (z.B. einem Münzenwurf mit Ausgang "Kopf" bzw. "Zahl"), das Mass 1 (gemessen in bit) zu

- Verallgemeinerung:
  Einem Zufallsereignis mit k gleich wahrscheinlichen Ausgängen ordnen wir für den Informationsgehalt das Mass

$$\log_2 k \quad = \quad -\log_2 (1/k) \; = \; -\log_2 p \qquad [\mathrm{bit}]$$

  zu, wobei $p = 1/k$ die Wahrscheinlichkeit eine Ausgangs ist.

- Begründung:
  Annahme: Jemand wählt zufällig eine Zahl zwischen 1 und k.
  Durch wieviele ja/nein Fragen kann diese Zahl bestimmt werden?

Einige Eigenschaften des Informationsgehalts

Der Informationsgehalt $\mathrm{h}(p)$

- ist unabhängig von der Codierung
- steigt wenn die Wahrscheinlichkeit p sinkt
  (da die Unsicherheit über den Ausgang des Zufallsexperiments steigt)
- ist "additiv": $\mathrm{h}(p_1 \cdot p_2) = \mathrm{h}(p_1) + \mathrm{h}(p_2)$

- immer gleiche Meldung : $\mathrm{h}(1) = 0$
- Meldung kommt nie vor: $\mathrm{h}(0) = \infty$
- 2 gleich wahrscheinliche Meldungen: $\mathrm{h}(1/2) = 1$
- Informationsgehalt einer Dezimalziffer: $\mathrm{h}(1/10) \approx 3.32$

Pro Memoria: Rechnen mit Logarithmen …

… hier mit dem Logarithmus Dualis ld (bzw. $\log_2$)

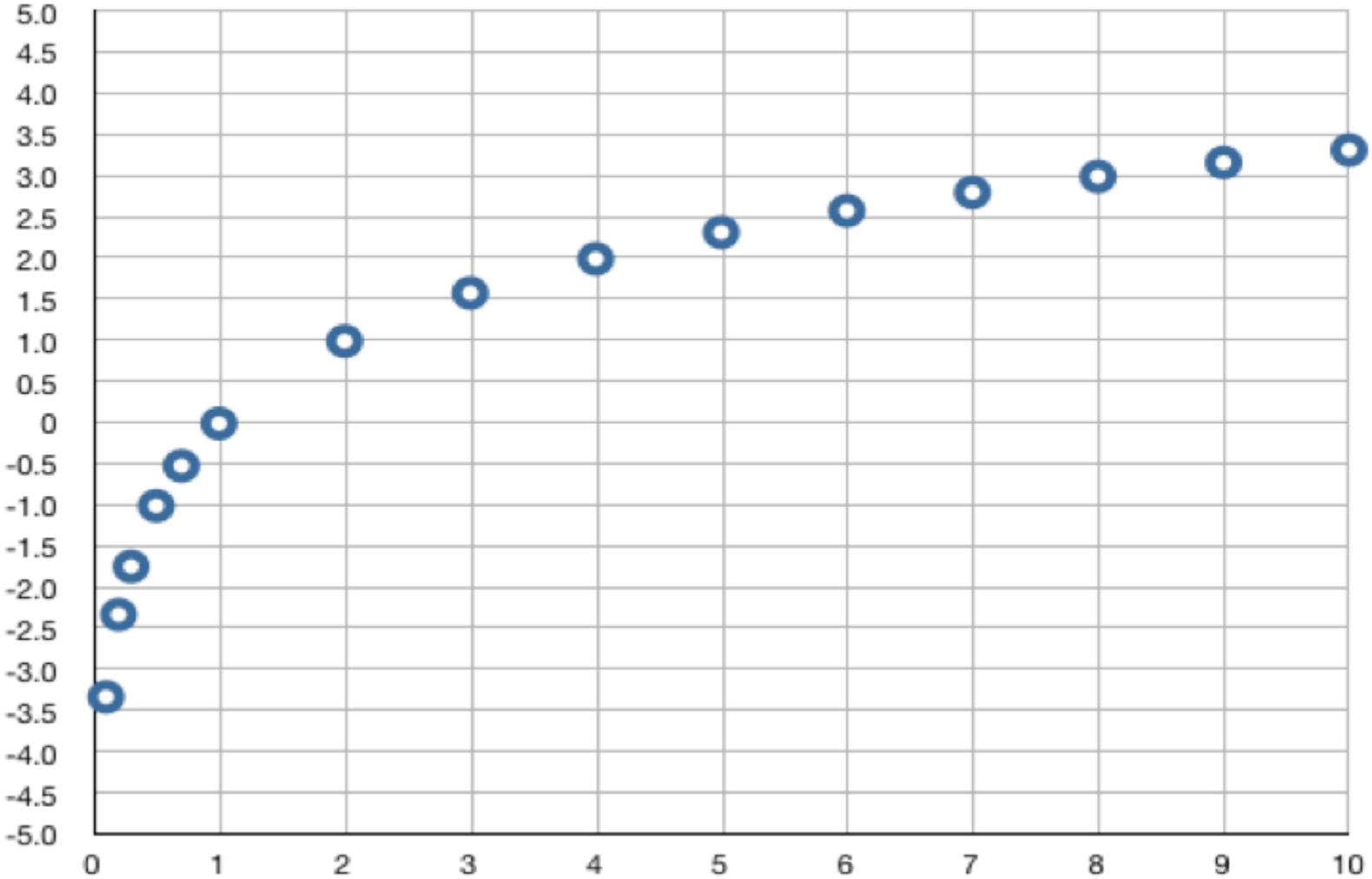
- $2^{\text{ld } x} = x$
- ld 1/x = - ld x
- ld (x*y) = ld x + ld y
- ld 2*x = 1 + ld x
- ld x^n = n*ld x
- ld x^2 = 2*ld x
- ld x = ln x / ln 2 ~ 1.44*ln x
- ld x = log x / log 2 ~ 3.32*log x

log = 10-er Logarithmus, ln = logarithmus naturalis

Wertetabelle des Logarithmus Dualis

1	0.00
2	1.00
3	1.58
4	2.00
5	2.32
6	2.58
7	2.81
8	3.00
9	3.17
10	3.32

Graph des Logarithmus Dualis



Entropie: Messen des mittleren Informationsgehalts

- Was geschieht, wenn nicht alle Ausgänge des Zufallsereignisses gleich wahrscheinlich sind?

- Die (individellen) Informationsgehalte der k verschiedenen Ausgänge des Zufallsereignisses müssen mit ihrer jeweiligen Wahrscheinlichkeit $p_i \ (1 \leq i \leq k)$ gewichtet werden:

$$\mathrm{H}(p_1, p_2, \ldots, p_k) = -(p_1 \cdot \log_2 p_1 + p_2 \cdot \log_2 p_2 + \ldots + p_k \cdot \log_2 p_k) \quad [\text{bit}]$$

bzw.

$$H = -\sum_{i=1}^{k} p_i \log_2 p_i \quad [\text{bit}]$$

- Der mittlere Informationsgehalt (average information content) wird auch als Entropie (entropy) bezeichnet.

Redundanz

- Mittlerer Informationsgehalt (Entropie):

$$H = -\sum_i p_i \cdot \log_2 p_i \qquad \text{[bit]}$$

- Mittlere Wortlänge:

$$L = -\sum_i p_i \cdot l_i \qquad \text{[bit]}$$

  wobei l_i die Wortlänge der Repräsentation des i-ten Ausgangs des Zufallsexperiments ist

- Redundanz:

$$R = L - H \qquad \text{[bit]}$$

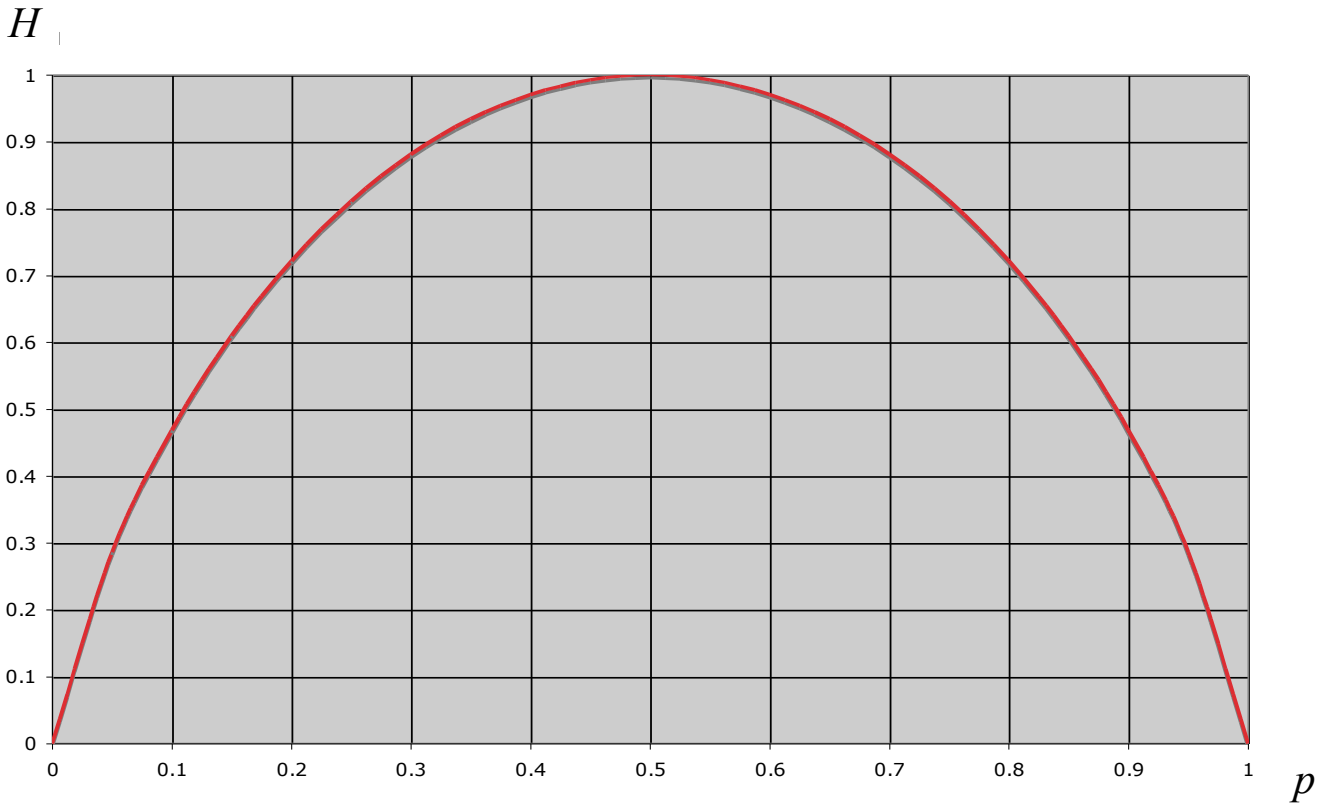
- Satz von Shannon (Shannon's Coding Theorem):

$$L \geq H$$

Entropiefunktion bei 2 möglichen Ereignissen

- Ein Ereignis hat W'keit p , das andere die W'keit $(1-p)$
- Mittlerer Informationsgehalt (Entropie) in diesem Fall:

$$H = -\left(p \cdot \log_2 p + (1-p) \cdot \log_2 (1-p) \right) \qquad [\text{bit}]$$



Bem.:
$p \cdot \log_2 p = 0 \quad$ für $p = 0$

Beispiel: Dezimalziffern

alle Ziffern gleich wahrscheinlich

- Informationsgehalt

$$H = -\ 10 \cdot (\ 0.1 \cdot \log_2 0.1\ ) = -\log_2 0.1 = \log_2 10 \approx 3.32 \quad [\text{bit}]$$

- Wortlänge

$$L = \text{ceil}(\log_2 10) = 4 \quad [\text{bit}]$$

- Redundanz

$$R = L - H \approx 4 - 3.32 = 0.68 \qquad [\text{bit}]$$

Beispiel: (Gross-) Buchstaben

alle Zeichen gleich wahrscheinlich

- Informationsgehalt

$$H = \log_2 26 \approx 4.7 \quad [\text{bit}]$$

Zeichen gemäss empirischer Verteilung in der deutschen Sprache

- Informationsgehalt

$$H \approx 4.1 \quad [\text{bit}]$$

Bem.:

Bei Ausnutzen der Verteilung von Bigrammen (Buchstabenpaaren): $H \approx 3.5$ [bit]

Beispiel: Worte in der deutschen Sprache

10 Millionen Wörter mit unterschiedlichen Wahrscheinlichkeiten

- Informationsgehalt

$$H \approx 11.8 \quad [\text{bit}]$$

- mittlere Wortlänge

$$L \approx 5.7 \quad [\text{Buchstaben}]$$

Informationsfluss beim Lesen

- ca. 25 Buchstaben pro Sekunde [s]

  entspricht ca. 50 bit/s

In 60 Jahren kann ein Mensch ca. $3 \cdot 10^{10}$ bit aufnehmen

Auflösung des menschlichen Auges

- ca. 6 Megapixel

Speicherkapazität des Gehirns

- ca. 10^{12} bit

Erbinformation

- ca. 10^{10} bit

Beispiel für Komprimierung (1)

- Annahme:

  Alphabet bestehend aus Zeichen a, b, c und d

  mit Auftretens-Wahrscheinlichkeiten 0.1, 0.2, 0.3 und 0.4

- Typisches Encoding:

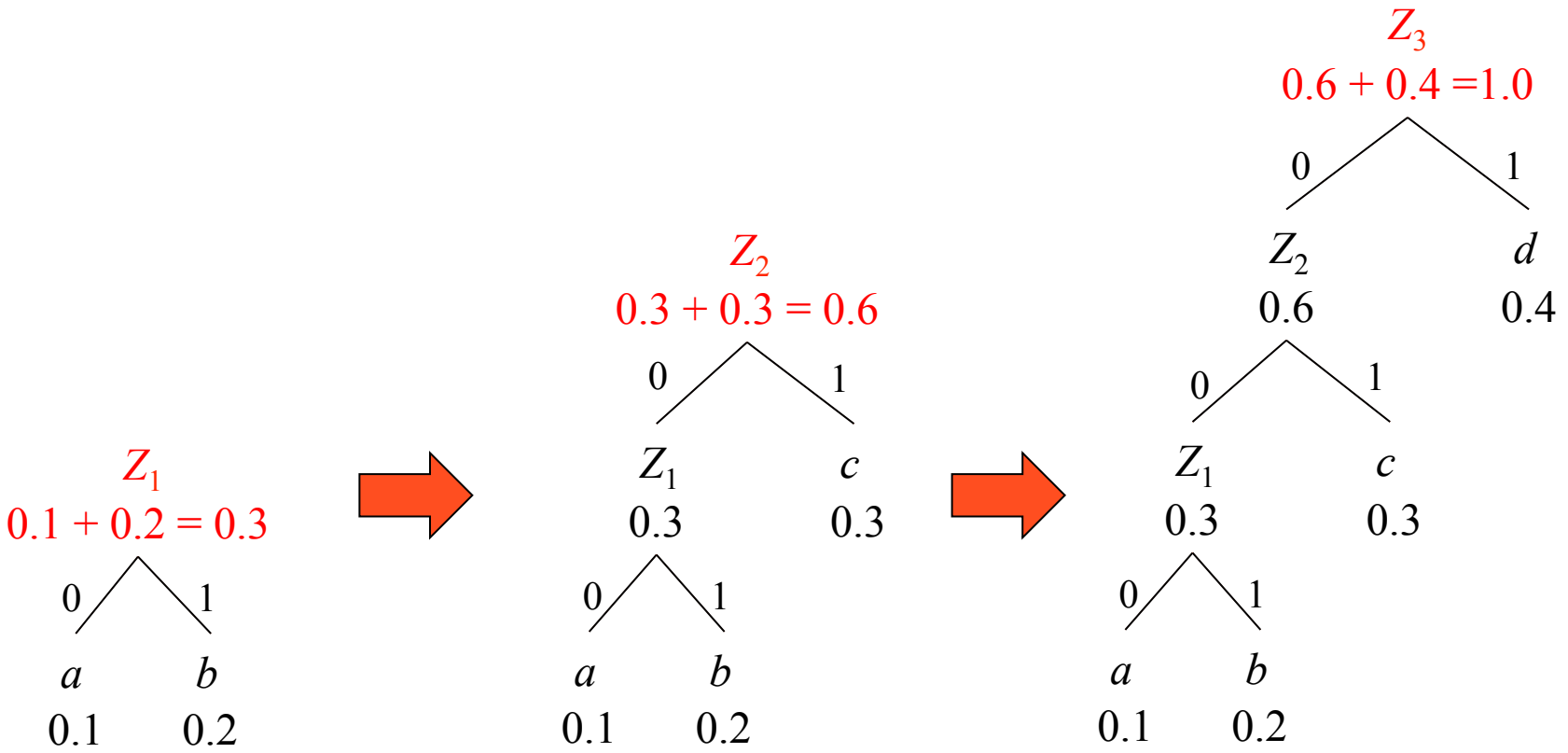
  Jeder Buchstabe mit 2 bit, z.B.

$$
\begin{aligned}
a &\rightarrow 00 \\
b &\rightarrow 01 \\
c &\rightarrow 10 \\
d &\rightarrow 11
\end{aligned}
$$

- Mittlere Wortlänge: 2 bit

Beispiel für Komprimierung (2)

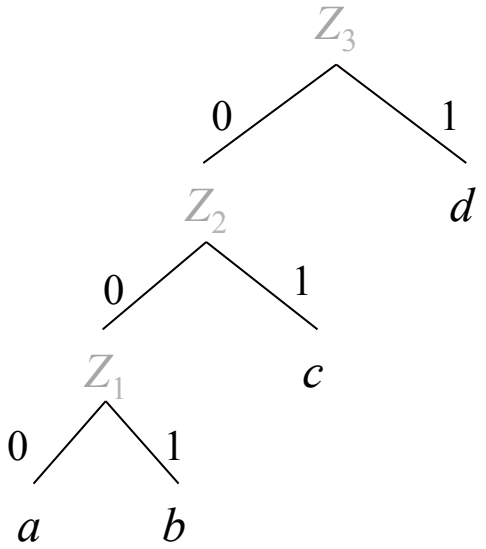
- Konstruktion eines Codebaums (Huffman Encoding):
  Die 2 seltensten Zeichen werden in einem "bottom up" Prozess schrittweise zu "Superzeichen" (Z_1 , Z_2 , Z_3) zusammengefasst



Beispiel für Komprimierung (3)

- Encoding: Tree Walk
  - Abstieg "nach links" ergibt binäre Ziffer 0
  - Abstieg "nach links" ergibt binäre Ziffer 1

$a \rightarrow 000$
$b \rightarrow 001$
$c \rightarrow 01$
$d \rightarrow 1$



- Decoding: Lesen einzelner bits
  1. bis zum ersten bit mit Wert 1
  2. maximal 3 bits
- Mittlere Worlänge

  1.9 bit (computation left as an exercise to the reader :-)