Elementare Angewandte Mathematik

Götz Kersting, WS 2016/17

Vorlesungsskript für L1, L2, L5-Studierende

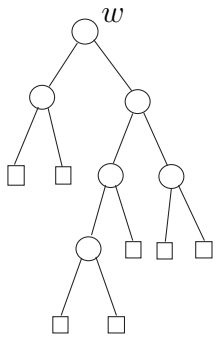
Inhaltsverzeichnis

1 Bäume

Bäume sind spezielle Graphen, die für praktische Zwecke besonders wichtig sind. Bei ihrer Untersuchung greifen wir auch auf Logarithmen zurück.

1.1 Binärbäume

Unter einem *Baum* verstehen wir hier einen Graphen der folgenden Gestalt.



Er besteht aus endlich vielen *Knoten*, die durch *Kanten* verbunden sind. Man spricht von einem Baum, weil sich keine geschlossenen ‚Rundwege' (‚Kreise') vorfinden. Wir betrachten hier nur Bäume, bei denen ein Knoten als *Wurzel* ausgezeichnet ist. Im Bild setzen wir ihn oben an die Spitze. Die Knoten werden dann unterschieden in *interne Knoten* (dargestellt als Kreise), und *externe Knoten* (dargestellt als Quadrate). Die externen Knoten nennt man auch *Blätter*. Aus ihnen wachsen nach unten keine weiteren Knoten. Wenn aus jedem internen Knoten nach unten höchstens 2 Kanten herausgehen, spricht man von einem *Binärbaum*. Wir betrachten in diesem Kapitel (wie im Bild) nur den Fall von genau 2 Kanten, den Fall von *vollen Binärbäumen*. Jeder Knoten k besitzt eine *Tiefe* $\ell(k)$, das ist die Anzahl der Kanten zwischen k und der Wurzel. Die Maximaltiefe der Knoten

$$h := \max_{k \in K} \ell(k)$$

heißt *Höhe* des Binärbaumes (K bezeichnet die Menge der Knoten). Im Beispiel ist $h = 4$.

Bemerkung. Da in der Graphentheorie auch Bäume ohne Wurzel betrachtet werden, wäre es genauer, hier von binären *Wurzelbäumen* zu sprechen. Wir behandeln in diesem Kapitel nur Bäume mit Wurzel.

Sei

$$m := \text{Anzahl der internen Knoten}\ , \quad n := \text{Anzahl der Blätter}$$

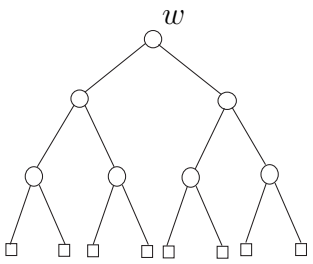
(im Beispiel $m = 6, n = 7$).

Satz. *Für jeden vollen Binärbaum gilt*

$$n = m + 1\ . \tag{1}$$

Beweis. Stellen wir uns vor, dass man einen vollen Binärbaum aus seiner Wurzel (oder irgend einem Knoten) heraus Schritt für Schritt ‚wachsen' lässt. Bei diesem Prozess verwandelt sich in jedem Schritt ein Blatt in einen internen Knoten, gleichzeitig wachsen aus diesem neuen internen Knoten zwei neue Blätter. In der Bilanz vergrößert sich die Anzahl der internen Knoten um 1, und genauso die Anzahl der Blätter um $2 - 1 = 1$. Die Differenz $n - m$ bleibt also unverändert. Sie beträgt am Anfang 1, dann ist die Wurzel externer Knoten, und $n - m = 1 - 0 = 1$. Daher gilt also, wie behauptet, immer $n - m = 1$. □

Das folgende Bild zeigt einen *vollständigen Binärbaum* der Höhe $h = 3$.



Er ist dadurch ausgezeichnet, dass alle Ebenen einer vorgegebenen Tiefe, bis zur letzten Ebene, vollständig mit Knoten aufgefüllt ist. Die Blätter haben alle Tiefe h, und die internen Knoten Tiefen kleiner als h.

Die Gesamtzahl der Knoten ist leicht ermittelt. Von einer Ebene zur nächsten verdoppelt sich die Knotenzahl, sie wächst also geometrisch, 1, 2, 4, 8 .... In der Tiefe t finden sich folglich 2^t Knoten. Die Anzahl der Blätter ist folglich $n = 2^h$ und nach (1) die Anzahl der internen Knoten $m = 2^h - 1$. Genausogut kann man die internen Knoten ebenenweise aufzusummieren und ihre Gesamtzahl als

$$m = 1 + 2 + 4 + \cdots + 2^{h-1}$$

bestimmen. Wir erhalten also mit unseren Ansatz nebenbei die Formel

$$1 + 2 + 4 + \cdots + 2^{h-1} = 2^h - 1 \ .$$

Man kann sie natürlich auch direkt beweisen, ohne binäre Bäume, etwa durch Induktion oder auch auf folgende Weise:

$$\begin{aligned} 1 + 2 + 4 + \cdots + 2^{h-1} &= (1 + 2 + 4 + \cdots + 2^{h-1}) \cdot (2 - 1) \\ &= (2 + 4 + 8 + \cdots + 2^h) - (1 + 2 + 4 + \cdots + 2^{h-1}) = 2^h - 1 \ . \end{aligned}$$

Satz. (Fano-Kraft) *Sei B die Menge der Blätter in einem vollen Binärbaum. Dann gilt*

$$\sum_{b \in B} 2^{-\ell(b)} = 1 \ . \tag{2}$$

Beweis. Stellen wir und vor, dass wir, ausgehend von der Wurzel, zufällig durch den Baum wandern, bis wir in einem Blatt landen. Der Weg wird durch Münzwurf gewählt, damit entscheiden wir, ob wir die rechte oder linke Kante nach unten laufen. Um zum Blatt b der Tiefe $\ell(b)$ zu gelangen, sind $\ell(b)$ Münzwürfe erforderlich, jedesmal mit dem „richtigen“ Resultat. Dies geschieht mit Wahrscheinlichkeit

$$\underbrace{\frac{1}{2} \cdot \frac{1}{2} \cdots \frac{1}{2}}_{\ell(b)\text{-mal}} = 2^{-\ell(b)} \ .$$

und diese Wahrscheinlichkeiten summieren sich zu 1 auf, da wir mit Wahrscheinlichkeit 1 in irgend einem Blatt landen. □

Wichtig (etwa in der Informationstheorie) ist auch die Umkehrung dieses Satzes: Zu jeder Folge $\ell_1, \ldots, \ell_n$ von natürlichen Zahlen mit

$$\sum_{i=1}^{n} 2^{-\ell_i} = 1$$

gibt es einen vollen Binärbaum mit n Blättern $b_1, \ldots, b_n$, so dass die Gleichungen $\ell(b_1) = \ell_1, \ldots, \ell(b_n) = \ell_n$ gelten, dass also $\ell_1, \ldots, \ell_n$ genau die Tiefen der Blätter im Baum sind. Man konstruiert den Baum einfach von der Wurzel aus.

1.2 Suchbäume

Aus der Informatik stammt folgendes Problem: m Nummern $a_1, \ldots, a_m$ sind so in einem Rechner abzuspeichern, dass man sie schnell wiederfinden kann. Man kann etwa an die Matrikelnummern von Studierenden denken und sich vorstellen, dass man am Speicherort mit einer Nummer auch direkten Zugriff zu anderen Daten der Person erhält.

Uns interessieren hier nicht die Details der Programmierung von Computern sondern eine idealisierte Situation. Wir betrachten binäre Bäume, an dessen Knoten die Nummern abgelegt sind, zusammen mit einer Suchstrategie, bei der die Suchzeit für eine Nummer (grob) der Tiefe des entsprechenden Knotens entspricht. Wir sehen die internen Knoten als Plätze für die Nummern vor, und die Blätter als Freistellen, in die zukünftig neue Nummern einsortiert werden können.

Idealerweise könnte man versuchen, die Nummern in einem *vollständigen* Binärbaum der Höhe h unterzubringen. Er hat viel Platz: Wie wir gesehen haben, hat er $m = 2^h - 1$ interne Knoten. So lassen sich in einem Baum der Höhe 20 immerhin schon $2^{20} - 1 = 1.048.575$ Nummern unterbringen, und für jede Nummer beträgt dann die Suchzeit höchstens 19.

Vollständige Bäume sind aber nur als Ideal nützlich. Es bleibt ungeklärt, wie sich darin die Nummern Einsortieren lassen, so dass man sie schnell wiederfindet. Wir betrachten nun *volle* binäre Bäume als Sortierschema. Seien $a_1, \ldots, a_m$ paarweise verschiedene Zahlen. Der zugehörige Suchbaum wird schrittweise nach einem Verfahren aufgebaut, bei dem die größeren Zahlen weiter rechts und die kleineren Zahlen weiter links untergebracht sind. Dann kann man offenbar eine gesuchte Zahl schneller wiederfinden. Dies erfordert, dass man zwischen Kanten unterscheidet, die von einem Knoten entweder nach rechts unten oder nach links unten abzweigen. Die Bilder machen dies evident (der Fachausdruck in der Mathematik ist *geordneter* Baum oder *planarer* Baum).

Genauer geht man folgendermaßen vor:

1. Plaziere a_1 an der Wurzel. Ergänze diesen Knoten in Tiefe 1 mit zwei Blättern. Sie sind mögliche Plätze für a_2.

2. Gilt $a_2 > a_1$, so gehe von der Wurzel nach rechts unten, lege a_2 in dem erreichten Blatt ab und verwandle dies zu einem neuen internen Knoten, indem man am Knoten nach unten zwei neue Blätter der Tiefe 2 anfügt. Gilt $a_2 < a_1$, so gehe stattdessen nach links unten und verfahre analog.

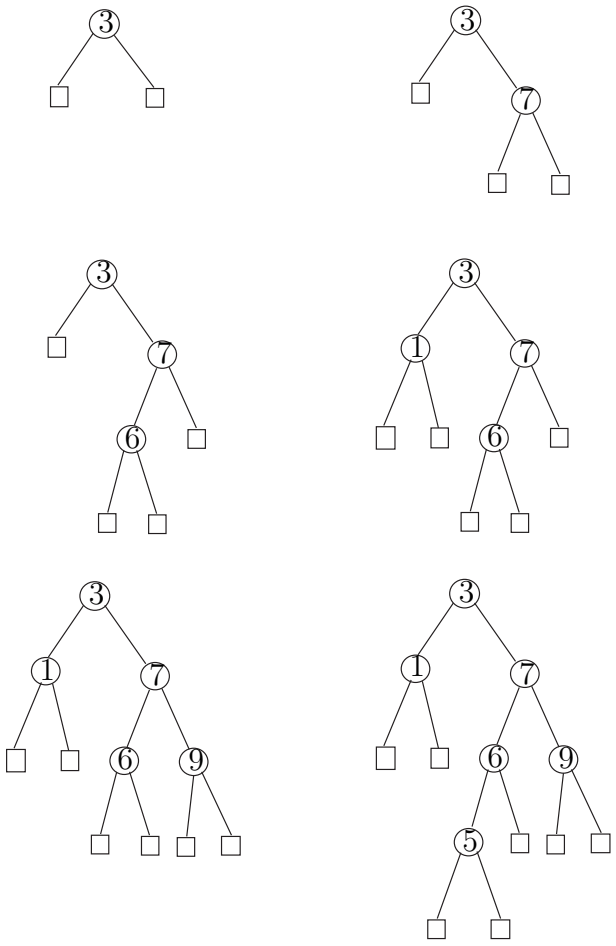
3. Allgemeiner: Sind schon $a_1, \ldots, a_{j-1}$ im Baum untergebracht, so durchlaufe den Baum von der Wurzel aus nach unten, bis ein Blatt erreicht ist.

An jedem besuchten internen Knoten vergleiche a_j mit der dort abgelegten Zahl a. Im Fall $a_j > a$ setzte den Weg nach rechts unten fort, im Fall $a_j < a$ nach links unten. Lege a_j im erreichten Blatt ab und mache es zum internen Knoten, indem man es nach unten durch zwei Blätter ergänzt.

Man bemerke, dass die gefundenen Bäume davon abhängen, in welcher Reihenfolge die Zahlen stehen. Jeder interne Knoten ist mit einer Zahl versehen, die Blätter sind freie Plätze, in denen nachkommende Zahlen einsortiert werden können.

Die Suche nach einer Zahl a funktioniert wie das Einsortieren. Man vergleicht a erst an der Wurzel mit a_1 und geht nach rechts oder links in die Tiefe, je nachdem ob $a > a_1$ oder $a < a_1$ gilt. So stößt man schließlich auf a (oder aber man stellt fest, dass sie gar nicht im Baum gespeichert ist).

Die folgenden Bilder veranschaulichen den Prozess für die Zahlen $3, 7, 6, 1, 9, 5$ (in dieser Reihenfolge!).



Offenbar sind diese Suchbäume im Allgemeinen nicht mehr vollständige sondern nur noch volle binäre Bäume. Eine interessante Frage ist es, die Bäume mit „möglichst vollständigen“ Bäumen der Höhe h zu vergleichen, die dieselbe Anzahl m von inneren Knoten hat. Die $n = m + 1$ Blätter eines solchen Baumes haben alle die Tiefe h oder $h - 1$, also gilt $2^{h-1} < n \leq 2^h$ bzw.

$$h - 1 < \log_2 n \leq h \ .$$

$\log_2 n$ ist also der Richtwert.

Mit diesem Wert wollen wir die Höhe von Suchbäumen vergleichen. Dazu machen wir eine *Modellannahme*, die besagt, dass die Zahlen eine rein zufällige Anordnung haben: Wenn bereits $j - 1$ Zahlen im Baum einsortiert sind, besitzt er $j - 1$ interne Knoten und j Blätter. Genauso gibt es zwischen den $j - 1$ Zahlen, auch rechts und links von ihnen, j Positionen, in denen sich eine neue Zahl einordnen kann. Die Blätter und die Positionen entsprechen einander. Wir nehmen nun an, dass a_j jede dieser j Positionen mit derselben Wahrscheinlichkeit einnimmt. (Man darf sich das auch so vorstellen, dass jede Reihenfolge von $a_1, \ldots, a_m$ dieselbe Wahrscheinlichkeit hat.) Im j-ten Schritt wird also jedes Blatt mit derselben Chance zum internen Knoten.

Wir haben es nun also mit zufälligen Suchbäumen zu tun. Zu seiner Beurteilung betrachten wir

$$\tau_n \ := \ \text{mittlere Tiefe eines Blattes im Suchbaum mit } n \text{ Blättern}$$

In Formeln ausgedrückt ist die mittlere Blattiefe für einen festen Baum gleich

$$\frac{1}{n} \sum_{b \in B} \ell(b) \ ,$$

wobei $\ell(b)$ wieder die Tiefe des Blattes b bezeichnet. Für einen zufälligen Baum mit n Blättern sind auch deren Tiefen zufällig, und man geht dann zum Erwartungswert über.

Beispiele. Für $n = 2$ gibt es gibt es nur einen Suchbaum, in dem beide Blätter die Tiefe 1 haben. Im Fall $n = 3$ gibt es für den Baum zwei Möglichkeiten, in beiden Fällen ist die mittlere Blatttiefe $(1 + 2 + 2)/3 = 5/3$.

Im Fall $n = 4$ muss man zwei Fälle unterscheiden. Nehmen wir an, dass wir die Zahlen 1,2,3 einordnen. Für die Reihenfolgen 2,1,3 und 2,3,1 einsteht ein vollständiger Binärbaum, bei dem alle Blätter die Tiefe 2 haben. Für die restlichen vier Reihenfolgen hat man je ein Blatt in Tiefe 1 und 2 und zwei

Blätter der Tiefe drei, also eine mittlere Blatttiefe $(1+2+3+3)/4 = 9/4$. Die erwartete mittlere Blatttliefe ist also

$$\tau_4 = \frac{2}{6} \cdot 2 + \frac{4}{6} \cdot \frac{9}{4} = \frac{13}{6} \qquad \square$$

Ein Schema ist in diesen Zahlen nicht zu erkennen, um so bemerkenswerter ist der folgende Satz.

Satz. $\tau_n = 2\Big(\frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n}\Big)$ *für* $n \geq 2$.

Beweis. Wir führen den Beweis per Induktion, den Induktionsanfang haben wir schon behandelt. Für den Induktionsschritt $n \to n+1$ betrachten wir erst einen festen vollen Binärbaum mit n Blättern der Tiefen $t_1, \ldots, t_n$. Es ist praktisch, erst mit der Gesamttiefe der Blätter

$$G_n = t_1 + \cdots + t_n$$

statt mit seiner mittleren Tiefe G_n/n zu rechnen. Der Baum hat $n-1$ innere Knoten, in die $n-1$ man Zahlen einsortieren kann, und n Blätter. Nehmen wir nun an, dass eine neue n-te Zahl einsortiert wird, die das i-te Blatt besetzt. Dann wird dieses Blatt zum inneren Knoten, an das wir eine Gabel mit zwei neuen Blättern anhängen. Beide haben die Tiefe t_i+1, während ein Blatt der Tiefe t_i verschwindet. Der neue Baum hat $n+1$ Blätter, seine Gesamttiefe ist

$$t_1 + \cdots + t_{i-1} + t_{i+1} + \cdots + t_n + 2(t_i + 1) = G_n + t_i + 2$$

Unsere Zufallsannahme über die Reihenfolge der einzuordnenden Zahlen bedeutet, dass die neu einsortierte Zahl mit derselben Chance in jedes der n Blätter gelangt, man muss also über $i = 1, \ldots, n$ mitteln, und wir erhalten als erwartete Gesamttiefe

$$G_{n+1} = G_n + \frac{1}{n}(t_1 + \cdots + t_n) + 2 = \frac{n+1}{n} G_n + 2$$

bzw.

$$\frac{1}{n+1} G_{n+1} = \frac{1}{n} G_n + \frac{2}{n+1} \ .$$

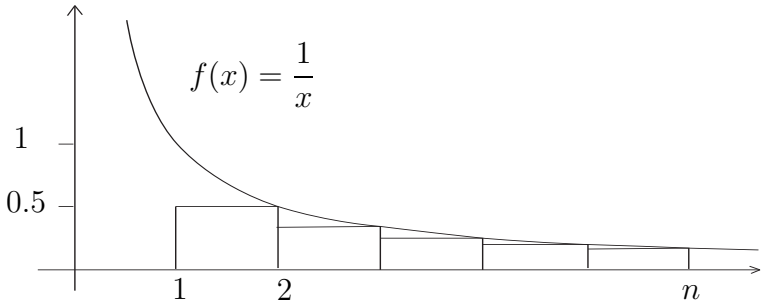
Die mittlere Blatttiefe vergrößert sich also um den festen Wert $2/(n+1)$. Dies Resultat überträgt sich unmittelbar auf zufällige Suchbäume, es folgt also wie behauptet

$$\tau_{n+1} = \tau_n + \frac{2}{n+1} = 2\Big(\frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n+1}\Big). \qquad \square$$

Weiter gilt

$$\frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n} \leq \int_1^n \frac{1}{x}\, dx = \ln n \ ,$$

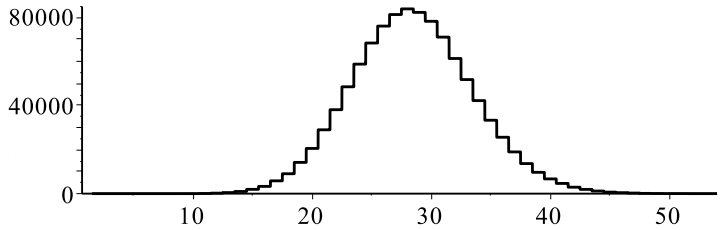
wie man dem folgenden Bild entnimmt.



Außerdem gilt $\ln n = \ln 2 \cdot \log_2 n = 0,69 \cdot \log_2 n$. Es folgt, dass in einem Suchbaum die mittlere Tiefe der Blätter logarithmisch mit der Anzahl der Blätter wächst:

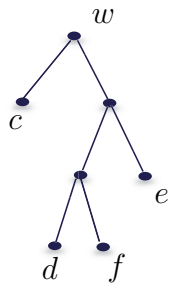
Korollar. $\tau_n \leq 1,4 \cdot \log_2 n$.

Zufällige Suchbäume haben also im Vergleich zu vollständigen Binärbäumen eine nur um den Faktor $1,4$ größere mittlere Tiefe. Dies belegt die Effizienz des beschriebenen Verfahrens. Die Graphik zeigt das Profil eines zufällig erzeugten Suchbaumes mit $n = 2^{20} = 1.048.576$ Blättern. Es zeigt die Anzahl der Blätter in Abhängigkeit von ihrer Tiefe. Das Maximum liegt bei $28 = 1,4 \cdot 20$.



1.3 Huffman-Codes

Jetzt betrachten wir endliche Mengen $\mathcal{A}$ und zugehörige Binärbäume, wobei jedem Element $a \in \mathcal{A}$ ein Blatt zugewiesen sei. Zum Beispiel kann man für $\mathcal{A} = \{c, d, e, f\}$ den folgenden Baum betrachten.



Der Weg von der Wurzel nach $a \in \mathcal{A}$ verläuft mal nach rechts, mal nach links, was sich in eine endliche Folge $k(a)$ von 1en und 0en übersetzt. Im Bild gilt $k(c) = 0, k(d) = 100, k(e) = 11, k(f) = 101$. Die Länge der Folge $k(a)$ bezeichnen wir mit $\ell(a)$, sie ist gleich der Tiefe des zugehörigen Blattes. Im Beispiel gilt $\ell(c) = 1, \ell(d) = 3, \ell(e) = 2, \ell(f) = 3$.

An folgende Interpretationen kann man denken.

1. $\mathcal{A}$ steht für ein Alphabet und der Binärbaum für einen digitalen Code, der a in den 01-string $k(a)$ überträgt. Solche Codes sind grundlegend für Computer. Weil nur Blätter Buchstaben tragen, ist keine Codierung eines Buchstaben ein Anfangstück der Codierung eines anderen Buchstaben. Solche Codes heißen ‚präfixfrei', sie erlauben eindeutige Entzifferung von ganzen Wörtern.

2. Alternativ lassen sich die Bäume als Fragestrategie auffassen, um ein Element $a \in \mathcal{A}$ mit Ja-Nein-Fragen zu identifizieren. Die erste Frage lautet „Fängt $k(a)$ mit einer 1 an?", die zweite „Ist die zweite Stelle von $k(a)$ eine 1?" etc. Dann ist $\ell(a)$ die Anzahl von Fragen, um a zu bestimmen.

3. Man kann auch mit mehr als zwei Symbolen arbeiten. Beim Morsen benutzt man drei (kurz, lang, Pause), dann wären ternäre Bäume anstelle binärer zu betrachten.

Treten die Buchstaben von $\mathcal{A}$ alle mit gleicher Häufigkeit auf, so wird man danach trachten, dass die Codierungen alle gleichlang sind. Anders verhält es sich, wenn manche Buchstaben bevorzugt auftreten, andere seltener, wie dies für Sprachen zutrifft. Wir nehmen nun an, dass jeder Buchstabe a eine Wahrscheinlichkeit p_a besitzt, mit der er eintritt. Für die deutsche Sprache

sind diese Wahrscheinlichkeiten ziemlich genau bekannt, z.B. hat das e eine Wahrscheinlichkeit von etwa 17%, das d von 5% und das x von weniger als 0.02%.

Zusammengefasst schreibt man $(p_a)_{a\in\mathcal{A}}$ für das System der Wahrscheinlichkeiten und nennt es eine *Wahrscheinlichkeitsverteilung* mit *Gewichten* p_a. Allgemein ist nur festzustellen, dass

$$p_a \geq 0 \ , \quad \sum_{a\in\mathcal{A}} p_a = 1$$

gilt.

Offenbar ist es sinnvoll, sich auf Codes zu beschränken, bei denen häufige Buchstaben kleine Codewörterlänge haben, für die genauer gesagt gilt:

$$p_a < p_b \quad \Rightarrow \quad \ell(a) \geq \ell(b)$$

für alle $a, b \in \mathcal{A}$. Hätte man nämlich Buchstaben a, b mit $p_a < p_b$ und $\ell(a) < \ell(b)$, so könnte man durch Vertauschen der Codewörter die Länge der Codierungen von a verlängert und die von b verkürzt, was günstig wäre.

Wir konstruieren nun optimale Codes. Damit meinen wir Präfixcodes k, die die erwartete Codewortlänge

$$E(k) := \sum_{a\in\mathcal{A}} \ell(a)p_a$$

minimieren. Sie heißen *Huffman-Codes* und werden von den Blättern hin zur Wurzel konstruiert. Sie sind von fundamentaler Bedeutung in der Informatik. Die Konstruktion von Huffman verfolgt man am besten an den Baumdarstellungen der Codes. Sie beruht auf den folgenden einfachen Feststellungen:

1. In einem optimalen Code verzweigt jeder innere Knoten (jeder Knoten, der kein Blatt ist) nach unten in *zwei* Kanten. Wäre da nur eine Kante, so könnte man sie aus dem Baum heraustrennen und erhielte damit verkürzte Codewortlängen.

2. In einem optimalen Code haben Buchstaben großer Wahrscheinlichkeit kurze Codierungen. Genauer: Aus $p_a < p_b$ folgt immer $\ell(a) \geq \ell(b)$. Andernfalls könnte man die Beschriftungen a und b im Baum vertauschen, wodurch sich die erwartete Codewortlänge um

$$\begin{aligned} &\ell(a)p_b + \ell(b)p_a - \ell(a)p_a - \ell(b)p_b \\ &\qquad = \ -\big(\ell(b) - \ell(a)\big)\big(p_b - p_a\big) < 0 \end{aligned}$$

verändern würde.

3. Sind also $u, v \in \mathcal{A}$ zwei Buchstaben kleinster Wahrscheinlichkeit,

$$p_u \leq p_v \leq p_a \quad \text{für alle} \quad a \neq u, v,$$

so folgt für einen optimalen Code

$$\ell(u) = \ell(v) \geq \ell(a) \quad \text{für alle} \quad a \neq u, v.$$

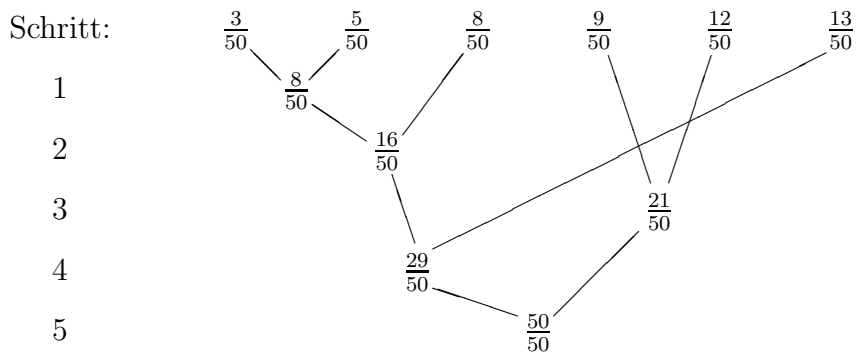
4. Damit dürfen wir nun auch annehmen, dass diese Buchstaben u und v in einem optimalen Code an derselben Gabel sitzen, d.h. dass sich die 01-Wörter $k(u)$ und $k(v)$ nur an der letzten Stelle unterscheiden. In dieser Situation kann man u, v zu einem neuen Buchstaben $\langle uv \rangle$ verschmelzen, zu dem kleineren Alphabet $\mathcal{A}' := S \cup \{\langle uv \rangle\} - \{u, v\}$ übergehen und dabei $\langle uv \rangle$ die Wahrscheinlichkeit $p_u + p_v$ zuweisen. Entsprechend kann man im Baum die u, v-Gabel beseitigen und an dem freiwerdenden Blatt den Buchstaben $\langle uv \rangle$ platzieren. Es ist offenbar: Der ursprüngliche Baum ist optimal für das ursprüngliche Alphabet, falls der reduzierte Baum für das reduzierte Alphabet optimal ist (die erwartete Wortlänge unterscheidet sich um $p_u + p_v$).

Es liegt nun auf der Hand, wir man von der Krone zur Wurzel einen optimalen Code erhält. Man verschmilzt erst die beiden Buchstaben kleinster Wahrscheinlichkeit. Man wiederholt diese Operation im reduzierten Alphabet und führt das Verfahren solange fort, bis alle Buchstaben verschmolzen sind. Nach diesem Verfahren entstehen Huffman-Codes.

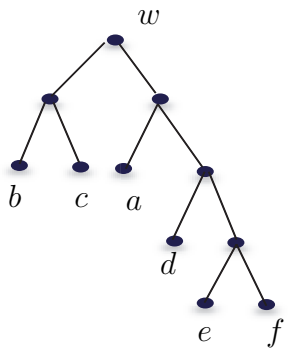
Beispiel. Wir führen das Verfahren exemplarisch für die Verteilung mit den Gewichten

$$p_a = \frac{13}{50},\ p_b = \frac{12}{50},\ p_c = \frac{9}{50},\ p_d = \frac{8}{50},\ p_e = \frac{5}{50},\ p_f = \frac{3}{50}$$

durch. Das folgende Schema zeigt die Reduktionsschritte.



Als Codebaum erhalten wir



Die mittlere Wortlänge berechnet sich als

$$E(k) = 4 \cdot \frac{3}{50} + 4 \cdot \frac{5}{50} + 3 \cdot \frac{8}{50} + 2 \cdot \frac{9}{50} + 2 \cdot \frac{12}{50} + 2 \cdot \frac{13}{50} = 2,48 \ . \qquad \square$$

Für große Alphabete ist die Konstruktion der Bäume und auch die Berechnung von $E(k)$ aufwendig. Man kann die erwartete Codewortlänge aber auch direkt aus den Wahrscheinlichkeitsgewichten p_a abschätzen. Dazu definieren wir die *Entropie* der Wahrscheinlichkeitsverteilung mit Gewichten $p_a, a \in \mathcal{A}$, als

$$H := -\sum_{a \in \mathcal{A}} p_a \log_2 p_a \ .$$

Im vorigen Beispiel ist

$$H = \frac{3}{50} \cdot 4,06 + \frac{5}{50} \cdot 3,32 + \frac{8}{50} \cdot 2,64 + \frac{9}{50} \cdot 2,47 + \frac{12}{50} \cdot 2,06 + \frac{13}{50} \cdot 1,94 = 2,443 \ .$$

Es gilt immer $H \geq 0$ (warum?).

Quellencodierungssatz. *Für die erwartete Codewortlänge $E(k)$ eines Huffman-Codes k gilt*

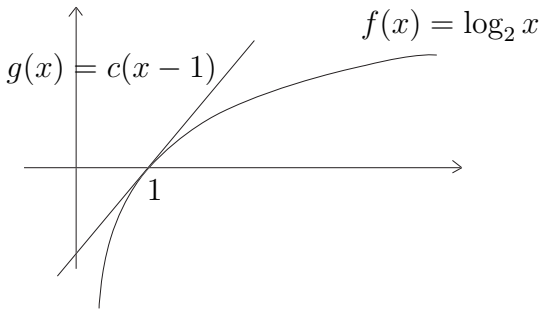
$$H \leq E(k) < H + 1.$$

Man kann also mit diesem Satz die erwartete Codewortlänge schnell und genau abschätzen, ohne den Baum überhaupt konstruieren zu müssen.

Beweis. Wir zeigen nur die (einfachere) Abschätzung nach unten. Es gilt

$$H - E(k) = \sum_a p_a(-\log_2 p_a - \ell(a)) = \sum_a p_a \log_2 \frac{2^{-\ell(a)}}{p_a} \ .$$

Nun liegt die Logarithmenfunktion unterhalb ihrer Tangenten $g(x)$ an der Stelle 1.



Folglich gilt $\log_2 x \le c(x-1)$ mit geeignetem $c > 0$, und

$$H - E(k) \le c \sum_a p_a \Big(\frac{2^{-\ell(a)}}{p_a} - 1 \Big) \le c \Big(\sum_a 2^{-\ell(a)} - 1 \Big) \ .$$

Schließlich gilt nach Fano-Kraft $\sum_a 2^{-\ell(a)} = 1$, wie wir im ersten Abschnitt festgestellt haben, und es folgt die $H \le E(k)$. □

Die „Quelle“ ist in der Informationstheorie der Ort, von wo die Nachrichten kommen. Dort benutzt man dann den Huffman-Code, um die Nachrichten in 01-Folgen zu transformieren. Daher rührt die Bezeichnung Quellencodierungssatz.

Bemerkung. Die übliche Interpretation der Entropie ergibt sich aus dem Quellencodierungssatz. Sei X ein „zufälliger Buchstabe“ aus dem Alphabet $\mathcal{A}$ mit Verteilung $p_a, a \in \mathcal{A}$, d.h. die Wahrscheinlichkeit, dass X gleich a ist, ist p_a. In Formeln:

$$\mathbf{P}(X = a) = p_a \ .$$

Stellen wir uns vor, dass wir X nicht kennen. Wie oben dargelegt können wir den Huffman-Code als Fragestrategie auffassen, um mit Ja-Nein-Fragen X von jemanden zu erfragen, der beobachten kann, welches der Buchstabe ist, den X in $\mathcal{A}$ angenommen hat.

Die Entropie gibt nach dem Quellencodierungssatz fast genau die mittlere Anzahl

$$E(k) = \sum_{a \in \mathcal{A}} \ell(a) \mathbf{P}(X = a)$$

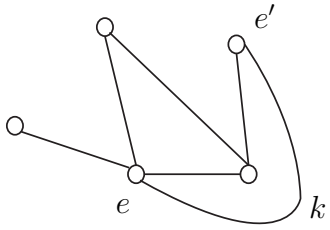
von Ja-Nein Fragen an, die dazu für uns notwendig ist. Dies ist gemeint, wenn man die Entropie beschreibt als den *Grad von Unbestimmtheit* oder *Ungewissheit* über den Wert, den X annimmt. Positiv ausgedrückt kann man auch vom *Informationsgehalt* des Zufallsexperiments sprechen, bei dem X entsteht. Dabei ist Information nicht inhaltlich gemeint, sondern in einem statistischen Sinn: Führt man ein Zufallsexperiment durch, bei dem „Erfolg“ mit Wahrscheinlichkeit p und „Misserfolg“ mit Wahrscheinlichkeit $q = 1 - p$ eintritt, so erfährt man wenig, wenn p nahe bei 0 oder 1 liegt, denn dann ist man sich über den Versuchsausgang schon von vornherein ziemlich sicher. So gesehen ist der Fall $p = 1/2$, der Wurf einer Münze, am informativsten. Da ist dann auch die Entropie maximal. – Die Informationstheorie, von Claude Shannon konzipiert, vertieft diese Interpretation und das Themengebiet der Nachrichtenübertragung.

2 Graphen

Die Theorie der Graphen hat eine geometrische Komponente, aber genauso eine algorithmische: Manche Fragen lassen sich rechnerisch schnell beantworten, andere nur mit allergrößtem Aufwand.

2.1 Kreise in Graphen

Unter einem Graphen verstehen wir hier (die Nomenklatur ist nicht einheitlich) ein Paar $G = (E, K)$, bestehend aus einer *Eckenmenge* E und einer *Kantenmenge* K. Das folgende Bild mit 5 Ecken und 6 Kanten zeigt, was gemeint ist.



Die Elemente aus E heißen *Ecken* oder *Knoten*, im Bild z.B. e und e'. Die Elemente $k \in K$ schreibt man gern in der Gestalt $k = \{e, e'\}$ mit Ecken $e, e' \in E$, $e \neq e'$. Wir interpretieren k als *Kante* zwischen e und e'. e, e' heißen dann *Nachbarn*. Es muss nicht jedes Paar von Ecken durch eine Kante verbunden sein – ist dies der Fall, so sprechen wir von einem *vollständigen Graphen*. Genauer gesprochen handelt es sich hier um *ungerichtete Graphen*, denn den Kanten ist keine Richtung gegeben. Gibt es Schleifen oder mehrere Kanten zwischen zwei Ecken, so spricht man von einem *Multigraphen*.

Der *Grad* $g(e)$ einer Ecke e ist die Anzahl ihrer Nachbarn,

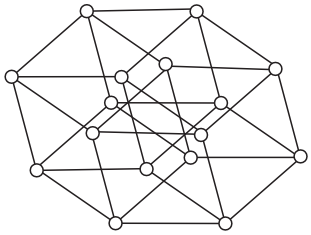
$$g(e) := \#\{e' : \{e, e'\} \in K\}.$$

Im Beispiel $g(e) = 4$, $g(e') = 2$.

Eine Folge $e_0, e_1, \ldots, e_l$ von Ecken heißt *Weg der Länge* l, falls e_{i-1}, e_i immer Nachbarn sind, falls also $\{e_{i-1}, e_i\} \in K$ für $i = 1, \ldots l$ gilt. Genauso kann man natürlich den Weg durch die Folge $k_1 = \{e_0, e_1\}, \ldots, k_l = \{e_{l-1}, e_l\}$ von Kanten gegeben denken. Gilt zusätzlich $e_0 = e_l$, handelt es sich also um einen geschlossenen Weg, so spricht man von einem *Kreis* (oder *Zyklus*).

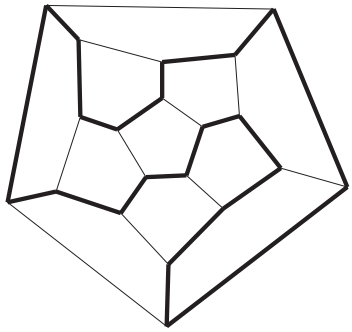
Ein Graph heißt *zusammenhängend*, wenn es zwischen je zwei Ecken e, e' immer einen Verbindungsweg gibt, also einen Weg $e_0, e_1, \ldots, e_l$ mit $e_0 = e$ und $e_l = e'$. Zusammenhängende Graphen ohne Kreise haben wir schon kennengelernt, sie heißen *Bäume*. (Jetzt betrachten wir nur unverwurzelte Bäume).

Beispiel: Würfel. Jeder Würfel ergibt einen Graphen mit 8 Ecken und 12 Kanten. Man kann ihn entstanden denken aus zwei Quadraten (Seitenflächen), deren Ecken passend verbunden werden. Nimmt man analog zwei Würfel und verbindet jede Ecke des einen Würfels mit der entsprechenden Ecke des anderen Würfels (im Bild sind das etwa alle waagerechten Kanten), so entsteht ein Hyperkubus „der Dimension 4“:



Aus zwei Hyperkuben der Dimension 4 entsteht analog einer der Dimension 5 und so weiter. □

Definition. Ein *Eulerkreis* ist ein Kreis, der jede Kante genau einmal passiert. Ein *Hamiltonkreis* ist ein Kreis, der jede Ecke genau einmal besucht.



ein Hamiltonkreis durch das ‚Dodekaeder‘

Die Fragestellungen, ob ein Graph einen Eulerkreis bzw. einen Hamiltonkreis enthält, erscheinen auf den ersten Blick wenig unterschiedlich. Das täuscht. Für Eulerkreise gibt es ein einfaches Kriterium.

Satz. *Ein Graph (und allgemeiner Multigraph) G enthält genau dann einen Eulerkreis, wenn er zusammenhängend ist und der Grad einer jeden Ecke eine gerade Zahl ist.*

Beweis. Die Bedingungen sind offenbar notwendig: Existiert ein Eulerkreis, so auch ein Verbindungsweg zwischen je zwei Ecken. Außerdem muss

der Eulerkreis, wenn er eine Ecke e besucht, sie auch wieder verlassen. Dies geschieht über verschiedene Kanten, damit hat e dann zwei Nachbarn. Besucht der Eulerkreis e zweimal, so hat e vier Nachbarn usw.

Die Bedingung ist auch hinreichend: Man wähle für den zu konstruierenden Eulerkreis einen Startpunkt e und bilde (nach irgendeiner Regel) einen Weg durch G, der keine Kante doppelt durchläuft. Wegen der Gradbedingung kann jede Ecke $e' \neq e$ wieder verlassen werden. Man gelangt also irgendwann nach e zurück. Es entsteht ein Kreis K_1, der aber noch nicht alle Ecken durchlaufen haben muss. Gibt es eine Ecke e' außerhalb K_1, so gibt es auch einen Weg von e nach e', und damit einen Weg von einer Ecke von K_1 nach e', der keine Kreiskante mehr benutzt. Wieder wegen der Gradbedingung kann man diesen Weg zu einem neuen Kreis K_2 fortsetzen, ohne Kanten aus K_1 zu benutzen. Beide Kreise kann man zu einem einzigen zusammenfassen. Dies setzt man fort, bis alle Ecken erfasst sind. □

Den zweiten Teil des Beweises kann man auch als Algorithmus ausgestalten. Dies ist der *Hierholzer-Algorithmus* aus 1873, er ist ein schneller Algorithmus.

Bemerkung. Ein Weg durch einen Graphen, der alle Ecken genau einmal besucht, sich aber nicht zum Kreis schließt, heißt *Eulerweg* (Beispiel: Haus vom Nikolaus). Das Kriterium ist nun: Es gibt genau dann einen Eulerweg, der sich nicht schließt, wenn genau zwei Ecken des Graphen ungeraden Grad haben (Begründung?).

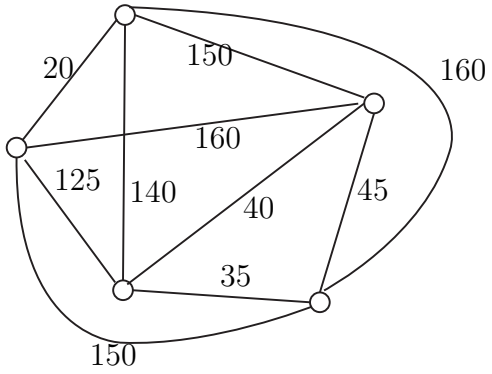
Für Hamiltonkreise gibt es kein ähnlich einfaches Kriterium. Dass ein Springer über ein Schachbrett hüpfen kann, so dass er jedes Feld genau einmal besucht und am Ende wieder im Ausgangspunkt zurückkehrt, ist nicht unmittelbar einsichtig; man findet Lösungen durch Ausprobieren. Auch gibt es keinen schnellen Algorithmus, um Hamiltonkreise zu finden. Im Gegenteil: Das Finden von Hamiltonkreisen ist ein Prototyp für Probleme, die algorithmisch besonders großen Aufwand erfordern. Würde man nämlich einen schnellen Algorithmus erfinden, so hätte man, wie sich herausstellt, auch schnelle Algorithmen für eine ganze Schar anderer notorisch schwieriger Probleme (alle sog. NP-vollständigen Probleme). Dass es solche Algorithmen nicht gibt, daran zweifelt heute noch kaum einer, auch wenn es bisher dafür keinen Beweis gibt. (Die Frage gehört zu den wichtigsten offenen Fragen der Mathematik.)

2.2 Das Problem des Handlungsreisenden

Wir kommen nun auf ein anderes notorisch schwieriges algorithmisches Problem zu sprechen, das *Problem des Handlungsreisenden* (traveling salesman problem, TSP). Jemand möchte eine Tour durch n Städte antreten und dann an den Ausgangsort zurückkehren. Die Kosten (Distanzen) für die Fahrt zwischen je 2 Städte sind gegeben (dabei seien die Kosten unabhängig davon, in welche Richtung man fährt, ob von A nach B oder B nach A. Man spricht deswegen genauer von einem symmetrischen TSP).

Aufgabe: Finde eine Rundtour mit minimalen Gesamtkosten.

Dieser Situation liegt ein Graph zugrunde.



Die Ecken sind die Städte, die Kanten die Verbindungswege dazwischen. Der Graph ist vollständig, d.h. je zwei Ecken sind direkt duch eine Kante verbunden. Jede Kante k besitzt ein *Gewicht* $d(k)$, das die Kosten (Distanz) für die Benutzung dieser Verbindungslinie angibt. Gesucht ist ein Hamiltonkreis $k_1, k_2, \ldots, k_n$ mit minimalen Gesamtkosten $d(k_1) + \cdots + d(k_n)$.

Ein genauerer Blick auf das Beispiel zeigt, dass hier der direkte Weg zwischen zwei Ecken immer der kürzeste ist. In Formeln ausgedrückt bedeutet dies

$$d(\{e_1, e_3\}) \leq d(\{e_1, e_2\}) + d(\{e_2, e_3\}) \quad (\textit{Dreiecksungleichung})$$

für beliebige Ecken e_1, e_2, e_3. Man spricht dann von einem *metrischen TSP*. Wir setzen dies im folgenden voraus.

Das Problem bei einem TSP ist nicht, einen Hamiltonkreis zu finden. In einem vollständigen Graphen ist das trivial, immer kann man eine beliebige, noch nicht besuchte Ecke ansteuern. Das Problem ist, dass so immens viele Hamiltonkreise existieren: Bei n Ecken hat man, ausgehend von einer Startecke, für die erste Fahrt $n-1$ Wahlmöglichkeiten, für die zweite dann noch $n-2$, usw. Insgesamt sind das

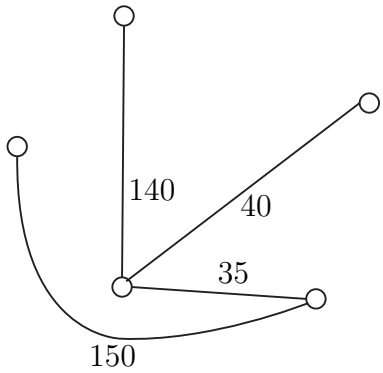
$$(n-1) \cdot (n-2) \cdots 2 \cdot 1 = (n-1)!$$

Hamiltonkreise. Für $n = 10$ sind das immerhin schon 362.880 verschiedene Rundfahrten! Dabei könnte man noch je zwei Touren zusammenfassen, die sich durch Umkehrung der Fahrtrichtung auseinander ergeben, aber das hilft auch nicht viel weiter.

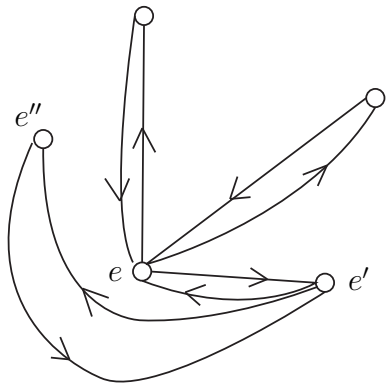
Um die optimale Rundtour zu finden, scheint kein besserer Algorithmus bekannt zu sein, als (im wesentlichen) alle Kreise durchzugehen. Verschiedene Strategien (die Informatiker sprechen von *Heuristiken*) wurden vorgeschlagen, um vergleichsweise gute Rundwege zu erhalten. Man kann etwa den Weg schrittweise aufbauen und dabei immer um eine möglichst kurze Strecke verlängern (im Beispiel: $20 + 125 + 35 + 45 + 150 = 375$). Man spricht dann von einem „greedy“, gierigen Verfahren. Eine andere Strategie besteht darin, von einer kürzesten Tour der Länge 3 auszugehen (im Beispiel $35 + 40 + 45$) und die schrittweise durch Hinzunahme einer weiteren Ecke zu immer größeren Kreisen auszudehnen, wobei pro Schritt der Längenzuwachs so klein wie möglich gehalten wird. Es stellt sich heraus, dass solche Heuristiken sehr schlechte Ergebnisse produzieren können.

Bemerkenswert ist, dass es schnelle Algorithmen gibt, die Rundwege produzieren, deren Länge *bis auf einen Faktor* optimal sind. Wir wollen ein solches Verfahren behandeln, bei dem dieser Faktor 2 ist. Damit erhält man also Rundtouren, deren Länge höchstens doppelt so lang wie die kürzeste Rundfahrt.

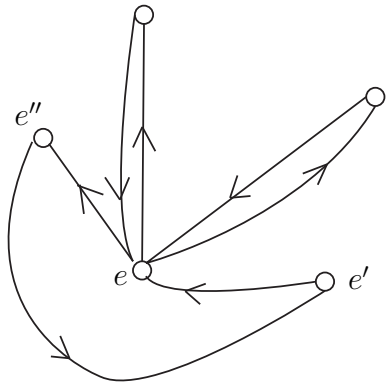
Für dieses Verfahren betrachtet man Spannbäume. Ein *Baum* ist ein zusammenhängender Graph, der keine Kreise besitzt. Ein in einem Graphen als Teilgraph enthaltener Baum heißt *Spannbaum*, wenn er alle n Ecken des Graphen enthält. Hier ist ein Spannbaum für obigen Graphen:



Jeder Spannbaum ergibt durch Verdoppeln jeder Kante einen Rundweg, bei die Ecken mehrfach besucht werden.



Jede mehrfach besuchte Ecke kann dann durch Wahl einer Abkürzung schrittweise zu einer einfach besuchten Ecke umgewandelt werden, ohne dass eine Ecke ausgelassen wird. Ist nämlich e, e', e'' ein Wegstück aus drei Ecken, und wird e' zweimal besucht, so kann man das Wegstück ohne weiteres durch e, e'' ersetzen, ohne das eine Stadt ausgelassen wird.



Bei einem metrischen TSP verkürzt sich dadurch die Tour. Die Gesamtlänge der am Ende resultierenden Rundtour mit Kanten $r_1, \ldots r_n$. ist höchstens so lang wie die die doppelte Summe der Gewichte im Spannbaum.

Diese Vorgehensweise ist natürlich besonders günstig, wenn wir von einem *minimalen Spannbaum* ausgehen, dessen Summe über die Kantengewichte also unter allen Spannbäumen am kleinsten ist. Wir können dann folgendermaßen abschätzen:

Seien $k_1, \ldots, k_n$ die Kanten einer optimalen Rundtour durch die n Ecken. Weil sie ein Kreis bilden, sind $k_1, \ldots, k_{n-1}$ die Kanten eines Spannbaums. d' sei die Summe der Kantengewichte dieses Baums.

Sei andererseits d'' die Summe der Kantengewichte eines minimalen Spannbaumes und $r_1, \ldots, r_n$ eine wie oben aus ihm resultierende Rundtour. Dann folgt nach unseren Überlegungen

$$d(r_1) + \cdots + d(r_n) \leq 2d'' \leq 2d' \leq 2(d(k_1) + \cdots + d(k_n)) \ .$$

Wir fassen zusammen:

Satz. *Eine aus einem minimalen Spannbaum erhaltene Rundtour hat eine Länge, die höchstens doppelt so lang ist wie diejenige einer kürzesten Rundtour.*

Diese Vorgehensweise ist natürlich nur dann von Nutzen, wenn es einen schnellen Algorithmus gibt, um minimale Spannbäume zu finden. Die Konstruktion solcher Algorithmen beruht auf folgenden Beobachtung für minimale Spannbäume S:

Proposition. Zerlegt man die Menge E der Ecken eines Graphen in zwei disjunkte Teile E_1 und E_2, $E = E_1 \cup E_2$, dann enthält jeder minimale Spannbaum S von G eine Kante $k = \{e_1, e_2\}$, die E_1 und E_2 verbindet und deren Gewicht $d(k)$ minimal ist unter allen Kanten $k' = \{e_1', e_2'\}$ mit $e_1' \in E_1$, $e_2' \in E_2$.

Beweis. Wir beweisen diese Aussage per Widerspruch. Andernfalls könnte man nämlich eine Kante k' finden, die E_1 und E_2 verbindet, die nicht zu S gehört und die ein kleineres Gewicht hat als alle Kanten aus S, die E_1 und E_2 verbinden. Fügt man k' zu S hinzu, so entsteht eine Kreis in S, der durch k' läuft. Dieser muss dann auch eine Kante k aus S zwischen E_1 und E_2 passieren. Beseitigt man nun k, so hat man einen neuen Spannbaum S' mit den Kantenmenge $K' = (K \setminus \{k\}) \cup \{k'\}$. Sein Gesamtgewicht ist kleiner als dasjenige von S. Dies steht im Widerspruch zur Minimalität von S. □

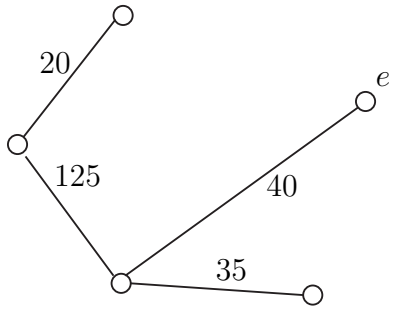
Dies führt unmittelbar zu einem „greedy“ Konstruktionsalgorithmus. Er ist von mehreren Autoren entdeckt worden und als *Algorithmus von Prim* bekannt.

Algorithmus für die Konstruktion eines minimalen Spannbaums. *Wähle einen Knoten e und die Kanten $k_1, k_2, \ldots$ des Baums schrittweise nach folgenden Regeln.*

1. *Wähle k_1 als eine Kante von kleinstem Gewicht, die den Knoten e enthält.*
2. *Sind $k_1, \ldots, k_{j-1}$ schon bestimmt, so wähle k_j als eine von den Kanten $k \neq k_1, \ldots, k_{j-1}$, so dass der von $k_1, \ldots, k_j$ gebildete Teilgraph ein Baum ist. Unter diesen Kante wähle eine von kleinstem Gewicht.*

Das Verfahren bricht ab, wenn ein Spannbaum entstanden ist. In unserem Beispiel werden, wenn man in der Ecke e ganz rechts im Graphen startet,

der Reihe nach die Kanten mit den Gewichten $40, 35, 125, 20$ ausgewählt, und der entstehende Spannbaum hat die Gestalt

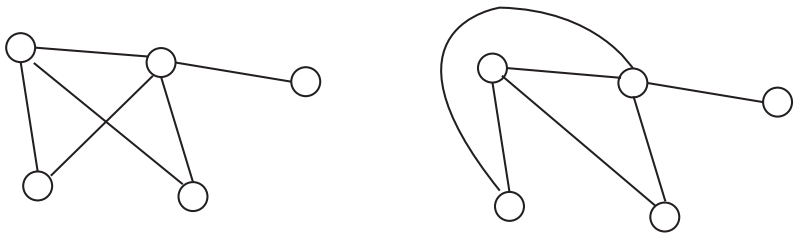


Offenbar ist der Algorithmus schnell und leicht durchzuführen. Auch ist es offensichtlich, dass er ein korrektes Resultat liefert, sofern alle Gewicht verschieden sind. Dann besteht nämlich keine Freiheit in der Wahl neuer Kanten, und der Spannbaum ist eindeutig bestimmt. Wenn die Gewichte einiger Kanten übereinstimmen, bleibt der Algorithmus korrekt (man kann diesen Fall auf den vorigen zurückführen, indem man die Gewichte der Kanten leicht verändert). Dann sind aber die Spannbäume im Allgemeinen nicht mehr eindeutig. Wir fassen zusammen:

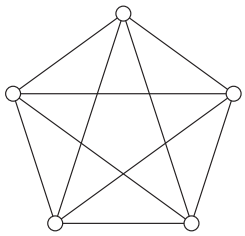
Satz. *Der Algorithmus von Prim zur Erzeugung eines minimalen Spannbaums in einem vollständigen, gewichteten Graph G ist korrekt.*

2.3 Planare Graphen

Wir wollen nun noch stärker geometrischen Aspekten von Graphen nachgehen. Das folgende Beispiel zeigt ein und denselben Graphen in zwei Darstellungen. Sie sind gleich, weil sie dieselben Anzahl von Ecken haben und entsprechende Paare von Ecken benachbart sind.

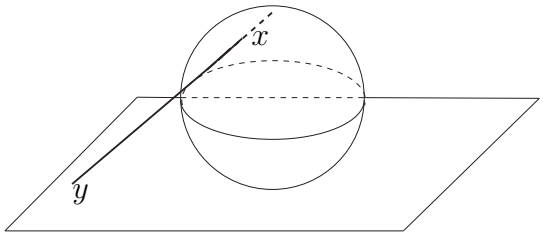


Im ersten kreuzen sich zwei Kanten, im zweiten ist das vermieden. Man kann sicher nicht immer erreichen, das es keine sich kreuzende Kanten gibt, aber wie ist das z. B. in der folgenden Situation (dieses ist der vollständige Graph K_5 mit 5 Ecken und allen möglichen Kanten)?



Definition. Ein Graph heißt *planar* (*plättbar*), falls sich seine Ecken und Kanten so in die Ebene legen lassen, dass sich keine zwei Kanten schneiden.

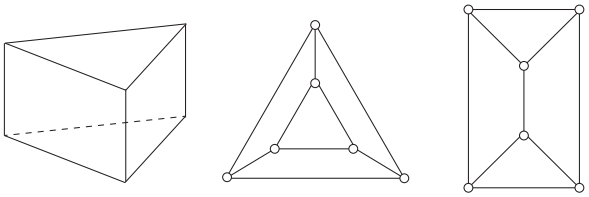
Genausogut kann man Graphen auf die Kugelsphäre zeichnen. Weil man aber Elemente x der Kugelsphäre (ohne den Nordpol n) und Elemente y der Ebene eins zu eine miteinander identifizieren lassen, macht das keinen wesentlichen Unterschied. Die Identifikation lässt sich mit der *stereographischen Projektion* bewerkstelligen. So kann man Ecken und Kanten eines Graphen in der Ebene auf eine Kugel übertragen, und umgekehrt.



Ob man einen planaren Graphen auf eine Sphäre zeichnet, oder in die Ebene, macht also keinen Unterschied.

Schließlich kann man die Ecken und Kanten eines *konvexen (nach außen gewölbten) Polyeder* (eines durch endlich viele ebene Flächen begrenzten Körpers – wie Würfel, Prismen, Oktaeder) kreuzungsfrei auf eine Kugel projizieren (auf Kantenlängen und Winkel kommt es dabei nicht an). So ergeben alle Polyeder im Raum planare Graphen in der Ebene. Solche Graphen nennen wir dann auch Polyeder. Sie sind zusammenhängend, jede Kante grenzt an genau zwei Flächen, jede Fläche hat mindestens drei Ecken und der Grad jeder Ecke ist mindestens gleich 3.

So lässt sich z.B. ein Prisma kreuzungsfrei in die Ebene einbetten. In der folgenden Illustration geschieht dies auf zweierlei Weise.



Ein planarer Graph zerlegt die Ebene in *Zellen* (Teilflächen), einschließlich einer äußeren, die den Graphen umfasst und die man immer mitzählt. Speziell bei der Überführung eines Polyeders in die Ebene entspricht sie einer Polyederfläche (im Prismenbeispiel ist das ein Dreieck bzw. ein Viereck).

Für planare Graphen gilt eine wichtige und berühmte Gesetzmäßigkeit, die die folgenden Größen zueinander in Beziehung setzt:

$f :=$ Anzahl der Zellen, einschließlich der äußeren,
$e :=$ Anzahl der Ecken,
$k :=$ Anzahl der Kanten.

Satz. Eulersche Polyederformel. *Für zusammenhängende planare Graphen gilt*

$$e + f = k + 2 \ .$$

Beweis. Ähnlich wie früher überzeugen wir uns von der Gültigkeit der Formel, indem wir einen planaren Graphen schrittweise aufbauen.

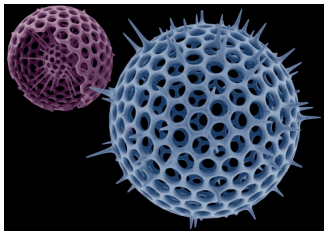
Bei einem zusammenhängenden Graphen mit zwei Ecken gilt $e = 2$, $f = 1$ und $k = 1$, so das die Polyederformel erfüllt ist.

Hängen wir nun schrittweise neue Kanten an den Graphen, so sind zwei Fälle möglich: Entweder wir benutzen dazu zwei vorhandene Ecken. Dann teilt die Kante eine vorhandene Zelle in zwei neue Zellen. Oder aber wir hängen eine Kante zusammen mit einer neuen Ecke an. Dann entsteht keine neue Zelle. In beiden Fällen erhöht sich k, wie auch $e + f$ um 1, und die Gültigkeit der Polyederformel bleibt erhalten.

So erreicht man, ausgehend von einer Kante, jeden planaren Graphen. □

Die Polyederformel ist grundlegend für die Untersuchung von Polyedern.

Beispiel. Golfbälle. Ein Golfball hat auf seiner Oberfläche 300 bis 450 kleine Dellen. Deren Ränder ergeben ein Polyeder aus (auf den ersten Blick) lauter Sechsecken, bei dem sich in jeder Ecke Ecke drei Flächen treffen. Ganz ähnlich ist die äußere Gestalt vieler Strahlentierchen (Radiolarien).



Ein genauerer Blick zeigt, dass dieses Schema an einzelnen Stellen durchbrochen ist und Fünfecke auftauchen. In der Tat ist es nicht möglich ein Polyeder nur aus Sechsecken zu bilden, so dass in jeder Ecke 3 Kanten zusammenstoßen: Jedes Fläche hätte dann 6 Kanten, und es folgt $k = 6f/2$, da jede Kante zwei Flächen begrenzt. Entsprechend folgt $e = 6f/3$, da jede Ecke 3 Flächen berührt. Wir erhalten

$$e + f = 2f + f = 3f = k$$

im Widerspruch zur Polyederformel. □

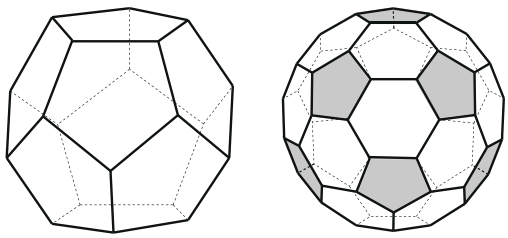
Beispiel. Fullerene. Wir betrachten nun Polyeder aus Fünf- und Sechsecken, bei denen sich in jeder Ecke drei Flächen treffen. In der Natur tauchen sie z. B. als Fullerene auf, das sind Moleküle nur aus Kohlenstoff. In den Ecken sind die Atome, entlang den Kanten wirken die chemischen Bindungen. Sind f_5 und f_6 die Anzahl der Fünf- bzw. Sechsecke

$$f = f_5 + f_6 \ , \quad 2k = 5f_5 + 6f_6 \ , \quad 3e = 5f_5 + 6f_6 \ .$$

Eingesetzt in die Polyederformel $6f + 6e = 6k + 12$ folgt

$$f_5 = 12 \ .$$

Man braucht genau zwölf Fünfecke. Die Anzahl der Sechsecke ist dagegen fast beliebig: Nur der Wert $f_6 = 1$ lässt sich nicht als Polyeder realisieren!



Beispiele sind das Dodekaeder und der Fußball, der erste Fall, bei dem keine Fünfecke aneinanderstoßen. □

Es ist anschaulich klar, dass ein planarer Graph nicht zu viele Kanten haben darf, sonst lässt er sich nicht mehr kreuzungsfrei in die Ebene legen. Für beliebige Graphen gilt $k \leq e(e-1)/2$, denn die Anzahl von Paaren von Ecken ist gerade $e(e-1)/2$. Dagegen gilt für einen planaren Graphen die folgende Ungleichung.

Satz. Für einen planaren Graphen mit $e \geq 3$ Ecken gilt $k \leq 3e - 6$.

Beweis. Bei einem planaren Graphen mit mindestens drei Ecken berührt jede Fläche mindestens drei Kanten (bis auf eine Ausnahme, welche?). Ist also s die Summe der Kantenanzahlen aller Flächen, so folgt $3f \leq s$. Dabei berücksichtigt man jede Kante höchstens zweimal, denn jede Kante stößt an höchstens 2 Flächen. Es folgt $s \leq 2k$ und damit

$$3f \leq 2k \ .$$

Mithilfe der Eulerformel folgt

$$k + 2 = e + f \leq e + \frac{2}{3}k \ ,$$

was die Behauptung ergibt. □

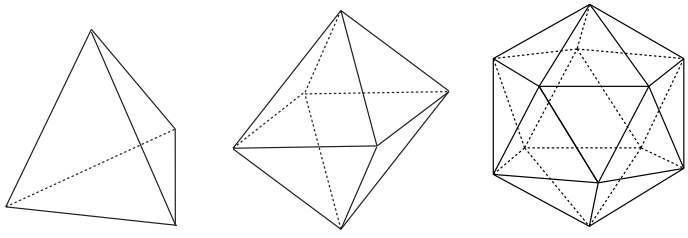
Beispiel. Ein nicht-planarer Graph. Der vollständige Graph K_5 mit $e = 5$ Ecken hat $k = 10$ Kanten. Er ist folglich nicht planar. □

Beispiel. Geodätische Kuppeln. Dies sind die Polyeder, deren Seiten nur aus Dreiecken bestehen. Sie werden in der Architektur benutzt um angenähert Kugeln bzw. (wie im Bild) Kuppeln herzustellen.

In diesen Polyedern gilt

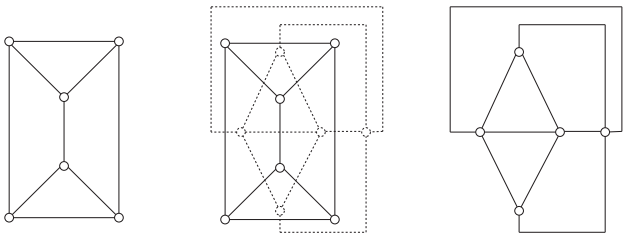
$$k = 3e - 6 \ ,$$

denn dann gehen im vorigen Beweis alle Ungleichungen in Gleichungen über. Bekannte Beispiele sind Tetraeder, Oktaeder und Ikosaeder.

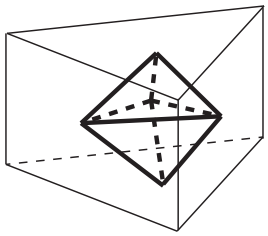


Durch Kappen der Ecken eines Ikosaeders in geeigneter Höhe gelangt man übrigens zum Fußball-Fulleren. □

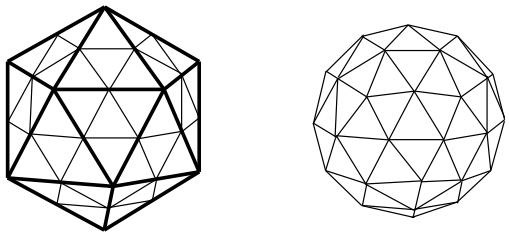
Zwischen Fullerenen und speziellen geodätischen Kuppeln lässt sich ein Zusammenhang herstellen. Dazu verwenden wir den Begriff des *dualen Graphen* $\tilde{G}$ eines planaren Graphen G. $\tilde{G}$ hat als Menge $\tilde{E}$ der Ecken $\tilde{e}$ gerade die Zellen von G, und zwei Ecken $\tilde{e}$ und $\tilde{e}'$ heißen benachbart, wenn die entsprechenden Zellen in g aneinandergrenzen. Für das Prisma sieht das so aus:



oder (räumlich gesehen)



Beispiele. In typischen Fällen sind geodätische Kuppeln so konstruiert, dass sich in jeder Ecke 5 oder 6 Kanten treffen. Dann erhält man Polyeder, die dual zu den Fullerenen sind. In Übertragung des Resultats aus dem letzten Beispiel erkennen wir, dass es dann genau 12 Ecken vom Grad 5 gibt. Das Duale des Dodekaeders ist das Ikosaeder mit 12 Ecken und 20 gleichseitigen Dreiecken. Wenn man seine Seitendreiecke in vier Teildreiecke zerlegt, entsteht ein Polyeder mit Eckengraden 5 oder 6.



Durch Absenken der zwölf Ecken des ursprünglichen Ikosaeders kommt man der Kugelgestalt nahe (die Seitendreiecke sind nun aber nicht mehr gleichseitig). Solche Methoden zur Konstruktion geodätischer Kuppeln hat man weit ausgebaut. □

Auch der Grad von Ecken ist bei planaren Graphen nicht beliebig.

Satz. Jeder planare Graph hat eine Ecke vom Grade höchstens 5.

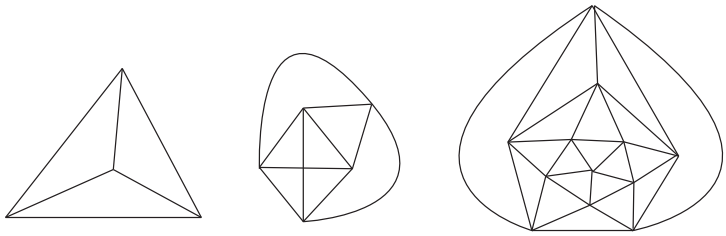
Beweis. Sei G ein Graph, dessen Eckengrad überall mindestens 6 ist. Für die Summe der Gradzahlen s folgt $s \geq 6e$. Da jede Kante zweimal zu s beiträgt, folgt $2k \geq 6e$ bzw. $k \geq 3e$. Daher kann G nach dem letzten Satz nicht planar sein. □

Beispiel: Färben von Landkarten. Karten färbt man so ein, dass benachbarte Länder verschiedene Farben haben. Wieviel Farben braucht man dazu? Wir fassen die Karte als planaren Graphen auf, dessen Flächen die Länder darstellen. Für den dualen Graphen G bedeutet dies, dass die Ecken gefärbt werden, und zwar so, dass benachbarte Ecken verschiedene Farben erhalten.

Wir färben nun nach folgenden Schema: Wir bestimmen in $G = G_1$ eine Ecke kleinsten Grades e_1 und entscheiden, dass sie als letzte gefärbt wird. Dann beseitigen wir sie samt aller Kanten, die von ihr ausgehen. Mit dem verbleibenden Teilgraphen G_2 verfahren wir analog, die dort gewählte Ecke e_2 wird als vorletzte gefärbt. Dies Vorgehen setzen wir fort, bis nur noch eine Ecke e_n übrig ist.

Es hat e_k in G_k höchstens den Grad 5. Färben wir also der Reihe nach $e_n, e_{n-1}, \ldots$, so kommen wir mit höchstens sechs verschiedenen Farben aus! Der berühmte 4-Farbensatz von Appel und Haken besagt, dass sogar vier Farben (aber nicht weniger) ausreichen. □

Beispiel. Regelmäßige Graphen. Wir nennen einen Graphen regelmäßig, wenn seine Ecken alle gleichviele Nachbarn haben, nämlich $p \geq 3$ Stück, und seine Zellen (einschließlich der äußeren) gleichviele begrenzende Kanten haben, nämlich $q \geq 3$ Stück. Beispiele geben die platonischen Polyeder: Würfel Tetraeder, Oktaeder, Dodekaeder und Ikosaeder.



Wir wollen zeigen, dass es keine weiteren gibt.

Da jede Kante zwei Ecken als Anfang und Ende hat und an zwei Flächen angrenzt, gilt

$$2k = pe \ , \quad 2k = qf \ .$$

Aufgelöst nach e und f und eingesetzt in die Polyederformel erhalten wir

$$\frac{2}{p}k + \frac{2}{q}k = k + 2 \ ,$$

und es folgt

$$\frac{2}{p} + \frac{2}{q} > 1.$$

Diese Ungleichung wird nur in den Fällen $p = q = 3$ oder $p = 4, q = 3$ oder $p = 3, q = 4$ oder $p = 5, q = 3$ oder $p = 3, q = 5$ erfüllt, sonst ist der linke Ausdruck, beginnend mit $p = q = 4$, kleiner oder gleich 1. Aus p und q berechnet sich dann aus der vorletzten Formel k und aus den beiden Formeln davor e und f. Dies entspricht genau den 5 Graphen, die wir bereits betrachtet haben. □

3 Codieren und Chiffrieren

Wir wenden uns nun stärker algebraisch ausgerichteten Teilen angewandter Mathematik zu.

3.1 Fehlerkorrigierende Codes

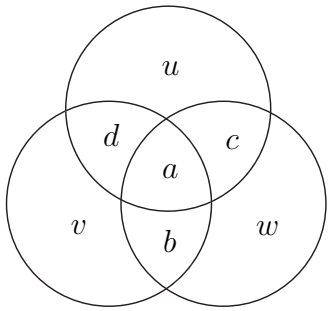
Wenn Nachrichten gesendet werden, können sich Übertragungsfehler einschleichen. Bei Texten bestehen da kaum Probleme, selbst bei vielen Fehlern ist man hinterher noch in der Lage, sie zu erkennen und zu beheben. Bei kodierten Nachrichten (etwa nach der Methode von Huffman) ist das Problem dagegen gravierend, und genauso bei der Übertragung von Daten. Moderne CD-Spieler etwa würden nicht funktionieren, wenn man nicht über *fehlerkorrigierende Codes* verfügte. Die Grundidee ist, dass man die Nachrichten redundant codiert, d.h. mehr Bits benutzt, als eigentlich (angesichts des Huffman-Codes) erforderlich wären.

Man unterscheidet zwischen *fehlererkennenden* und *fehlerkorrigierenden* Kodes. Ein fehlererkennender Code entsteht z. B., wenn man einer 01-Folge $a \dots d$ ein Prüfbit $e \in \{0, 1\}$ anhängen, so dass $s := a + \cdots + d + e$ eine gerade Zahl ist („parity check“). Ähnlich geht man z.B. bei der ISBN-Codierung von Büchern vor. Stellt man jedoch fest, dass s ungerade, also sicher ein Bit falsch ist, so kann man noch lange nicht den Fehler - allein anhand der codierten Nachricht - beseitigen. Solche Methoden sind etwa bei der Datenübertragung von Satelliten oder für CD-Spieler unbrauchbar.

Eine anderes Rezept zur Fehlererkennung ist, jedes Bit a doppelt zu übertragen: *aabbccdd* statt *abcd*. Fehlerkorrektur wird möglich, wenn man jedes Bit dreifach gesendet wird: Die empfangene Nachricht *acabbbccaddd* wird man dann als *abcd* entziffern, und nicht als *cbad*.

Hamming hat eine elegantere und effizientere Methode zur Fehlerkorrektur erfunden, die weite Anwendung findet und mit der wir uns nun eingehender befassen wollen. Sie ist ein $(7, 4)$-*Blockcode*, d.h. sie ergänzt jede Folge *abcd* aus 0en und 1en der Länge 4 zu einer 01-Folge *abcduvw* der Länge 7 mit drei zusätzlichen Prüfbits, so dass ein einzelner Fehler erkannt und korrigiert werden kann.

Das folgende Diagramm mit drei Kreisen erläutert das Vorgehen. Das Symbol a kommt in den Schnitt der 3 Kreise, b, c, d dorthin, wo je 2 Kreise sich schneiden. In den freigebliebenen Teilen der Kreise werden die Prüfbits u, v, w eingetragen, sie werden so gewählt, dass die Summe in jedem Kreis geradzahlig ist.



In Formeln: Zu $a, b, c, d \in \{0, 1\}$ wähle $u, v, w \in \{0, 1\}$, so dass

$$\begin{aligned} a + b + c + w &= \text{geradzahlig} \\ a + c + d + u &= \text{geradzahlig} \\ a + b + d + v &= \text{geradzahlig} \end{aligned} \tag{3}$$

Also: Für $abcd = 1001$ ist $uvw = 001$.

Der Clou ist, dass ein Fehler in der Reihe $abcduvw$ entdeckt und korrigiert werden kann. Dann sind manche dieser Summen ungeradzahlig: Bei a alle drei Summen, bei b, c, d jeweils 2 Summen und bei u, v, w jeweils eine Summe. Das falsche Bit lässt sich in jedem Fall ermitteln. Wir halten dies fest:

Satz. *Der Hammingcode kann einen Fehler korrigieren.*

Anders ausgedrückt: Ist $ABCDUVW$ eine beliebige 01-Folge der Länge 7, so gibt es genau eine Folge $abcduvw$, die Gleichungen (3) erfüllen und sich nur um höchstens 1 Bit von der vorgegebenen Folge unterscheidet. Sind etwa die Summen $A + B + C + W$, $A + C + D + U$, $A + B + D + V$ alle ungerade, so muss man A verändern, um die Gleichungen zu erfüllen.

Damit wird eine übersichtliche Struktur des Raumes $\{0, 1\}^7$ aller 01-Folgen der Länge 7 erkennbar. Insgesamt gibt es 2^7 derartige Folgen. Darunter sind 2^4 Folgen *Codewörter*. Sie entsprechen jeder möglichen Wahl von a, b, c, d. Um jedes Codewort gibt es eine „Umgebung“ benachbarter 01-Folgen, die sich um genau 1 Bit vom Codewort unterscheiden. Diese 7 Folgen werden bei Fehlerkorrektur in das Codewort übersetzt. Korrekte Übertragung führt natürlich auch zu korrekter Entschlüsselung, damit haben $7+1 = 8 = 2^3$ Folgen dieselbe Entschlüsselung. Alle 01-Folgen der Länge 7 sind erfasst, in der Tat gilt

$$2^4 \cdot 2^3 = 2^7 .$$

Damit ist jede Folge der Länge 7 entweder ein Codewort, oder es unterscheidet sich von genau einem Codewort an genau einer Stelle. Keine Folge wird dabei übersehen. Besser lässt sich der Raum der Folgen nicht aufteilen. Ein

Code, der in dieser Weise alle Folgen optimal ausschöpft, nennt man einen *perfekten Code*.

Der Gewinn lässt sich in einer einfachen Rechnung mit Wahrscheinlichkeiten erkennen. Wird ein Bit mit Wahrscheinlichkeit $p = 0,05$ vertauscht - von Bit zu Bit unabhängig -, so gibt der Hammingcode das richtige Wort mit Wahrscheinlichkeit

$$q^7 + 7pq^6 \;=\; 0,956\ ,$$

mit $q = 1 - p$. Dagegen wird ein Wort *abcd* ohne Korrekturmöglichkeit mit Wahrscheinlichkeit

$$q^4 \;=\; 0,81$$

richtig übertragen.

Die geometrische Darstellung des Hammingcodes mittels Schnitt von 3 Kreisen ist suggestiv, für den Rechner aber ungeeignet. Für ihn ist ein algebraischer Zugang besser geeignet, der das Rechnen mit geraden und ungeraden Zahlen stärker formalisiert. Wir beschreiben dies nun etwas genauer.

Die Menge $\mathbb{Z}$ der ganzen Zahlen zerfällt in die beiden Klassen $\mathbb{G}$ der geraden und $\mathbb{U}$ der ungeraden Zahlen. Wie üblich schreiben wir 0 für $\mathbb{G}$ und 1 für $\mathbb{U}$, betonen aber, dass nun mit 0 und 1 nicht mehr die gewöhnlichen ganzen Zahlen gemeint sind. In der Mathematik schreibt man dann für die Menge beider Klassen

$$\mathbb{Z}_2 \;=\; \{0,1\}\ .$$

Die arithmetischen Eigenschaften von $\mathbb{Z}_2$ sind folgendermaßen. Zwei gerade - oder zwei ungerade - Zahlen addiert ergeben eine gerade Zahl, eine gerade und eine ungerade Zahl addiert ergibt eine ungerade Zahl. Dies ergibt folgende Additionstabelle.

$+$	0	1
0	0	1
1	1	0

Genauso lassen sich gerade und ungerade Zahlen multiplizieren.

$\cdot$	0	1
0	0	0
1	0	1

Entscheidend ist, dass die üblichen Rechenregeln der ganzen Zahlen - Assoziativgesetz, Distributivgesetz, Kommutativgesetz - sich auf $\mathbb{Z}_2$ übertragen. Wer's nicht einsehen will, muss es nachrechnen.

Damit geht (3) in ein Gleichungssystem über, nämlich

$$\begin{aligned} a+b+c+w &= 0 \\ a+c+d+u &= 0 \\ a+d+b+v &= 0\ , \end{aligned}$$

ein ganz gewöhnliches, abgesehen davon, dass nun in $\mathbb{Z}_2$ gerechnet wird und nicht mehr in den rationalen Zahlen. Für diejenigen, die sich mit Matrizen auskennen, kann man das auch so schreiben:

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} a \\ b \\ c \\ d \\ u \\ v \\ w \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \tag{4}$$

Die Matrix heißt *Hammingmatrix*, man kann sich merken, dass sie als Spalten alle 01-Folgen der Länge 3 enthält, bis auf 000. Man bezeichnet sie üblicherweise mit H. Dass es beim Übertragen des Codewortes einen Übertragungsfehler gibt, kann man nun mittels Vektoraddition schreiben. Wird etwa a falsch übertragen, so gilt

$$\begin{pmatrix} A \\ B \\ C \\ D \\ U \\ V \\ W \end{pmatrix} = \begin{pmatrix} a \\ b \\ c \\ d \\ u \\ v \\ w \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Bei einem Übertragungsfehler enthält der Fehlervektor rechts genau eine 1, und zwar an der Stelle, wo der Fehler eingetreten ist. Unter Beachtung von (4) folgt

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} A \\ B \\ C \\ D \\ U \\ V \\ W \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}\ .$$

Der rechte Ausdruck ergibt eine Spalte der Länge 3, die sich in der Hammingmatrix genau dort befindet, wo der Übertragungsfehler steckt.

Zusammenfassend ergibt sich folgendes *Rezept zur Korrektur*: Multipliziere die Hammingmatrix H mit dem empfangenen Wort, aufgefasst als Spaltenvektor. Ergibt sich der Nullvektor, so nimm keine Korrektur vor. Ergibt sich ein anderer Vektor, so korrigiere an der Stelle, wo sich dieser Spaltenvektor in der Hammingmatrix befindet.

Man beachte, dass bei mehr als einem Übertragungsfehler das Verfahren nicht mehr funktioniert und sogar zusätzliche Fehler produzieren kann.

Zusammenfassend halten wir fest: Geometrisch dargestellt ist der Hammingcode suggestiv. Algebraisch kann ihn der Rechner besser verarbeiten. Er ordnet sich dann in den Rahmen der Linearen Algebra ein ($\{0,1\}^7$ wird zum Vektorraum der Dimension 7 mit Skalarbereich $\mathbb{Z}_2$) und erfährt auf diese Weise auch die Möglichkeit der Verallgemeinerung. Lineare Codes sind ein wichtiges Thema der Codierungstheorie.

3.2 Modulares Rechnen

Modulares Rechnen geht auf Gauß zurück. Hier behandeln wir es speziell mit Blick auf die Kryptographie.

Es geht um das Rechnen mit Resten. Man gibt sich eine natürliche Zahl $m > 1$, den „Modul“ vor und zieht von jeder ganzen Zahl a nur noch den Rest r in Betracht, der bei Division von a durch m,

$$a = um + r \ , \quad 0 \leq r < m \ ,$$

übrigbleibt. Zwei Zahlen a und b, die denselben Rest “modulo m“ haben, werden in dieser Sichtweise nicht mehr unterschieden. Dies ist genau dann der Fall, wenn $a - b$ ein Vielfaches von m ist, man nennt dann a und b *kongruent modulo* m und schreibt

$$a \equiv b \bmod m \ ,$$

etwa $10 \equiv 4 \bmod 3$ und $5 \not\equiv 3 \bmod 4$.

Entscheidend ist die Beobachtung, dass man modular im Großen und Ganzen genauso rechnen kann wie mit normalen Zahlen. Dies ermöglicht der folgende Satz.

Satz. *Sei $m > 1$ und $a \equiv b \bmod m$, $c \equiv d \bmod m$. Dann folgt*

$$a + c \equiv b + d \bmod m \ , \quad ac \equiv bd \bmod m \ .$$

Beweis. Nach Voraussetzung gilt $a - b = um$ und $c - d = vm$. Dann folgt

$$(a + c) - (b + d) = (u + v)m \ , \quad ac - bd = (a - b)c + b(c - d) = (uc + bv)m \ ,$$

und dies ergibt die Behauptung. □

Beispiele. Die folgenden Rechnungen beruhen auf dem vorigen Satz.

1. Wir wollen die beiden letzten Dezimalstellen von 24^{2008} berechnen. Dies bewerkstelligen wir, indem wir 24^{2008} modulo 100 berechnen. Es gilt $24^2 = 576 \equiv -24 \bmod 100$ und folglich (per Induktion) $24^k \equiv (-1)^{k-1} \cdot 24 \bmod 100$. Wir erhalten also $24^{2008} \equiv -24 \bmod 100$. Die letzten beiden Stellen von 24^{2008} sind also 76.

2. **Fibonacci-Zahlen.** Die Zahlen $F_1, F_2, \ldots$ gleich $1, 1, 2, 3, 5, 8, 13, 21, 34, \ldots$ sind so definiert, dass, beginnend mit $F_1 = F_2 = 1$, die Summe zweier aufeinanderfolgender Zahlen die nächsthöhere Zahl ergibt. In Formeln ausgedrückt: $F_n = F_{n-1} + F_{n-2}$, $n \geq 3$. Wir wollen zeigen, dass jede vierte dieser Fibonacci-Zahlen durch 3 teilbar ist. Dazu betrachten wir die Iteration modulo 3 und erhalten die Folge $1, 1, 2, 0, 2, 2, 1, 0, 1, 1, 2, 0, \ldots$ Das Schema wiederholt sich periodisch, wobei jede vierte Zahl den Rest 0 modulo 3 hat, also durch 3 teilbar ist.

3. **Neunerprobe.** Jemand hat $40752 \cdot 32111 = 1308587572$ ausgerechnet. Er bestimmt auch die Quersummen der drei Zahlen und erhält 0, 8 und 1. Wegen $0 \cdot 8 \neq 1$ erkennt er, dass ein Rechenfehler vorliegt. – Es handelt sich um Rechnen modulo 9. Es gilt $10 \equiv 1 \bmod 9$, also $40752 = 4 \cdot 10^4 + 7 \cdot 10^2 + 5 \cdot 10 + 2 \equiv 4 + 7 + 5 + 2 = 18 = 10 + 8 \equiv 1 + 8 = 9 \equiv 0 \bmod 9$ etc. Aus $40752 \cdot 32111 = 1308587572$ müsste also $0 \cdot 8 = 1$ folgen, ein Widerspruch.

4. **ISBN-Prüfziffern.** Die ISBN (international standard book number) eines Buches bestand vor dem Jahr 2007 aus 10 Ziffern $a_1, \ldots, a_{10}$. Unter ihnen ist die letzte eine Prüfziffer, die neben 0 bis 9 auch gleich X sein kann. Sie errechnet sich aus den Ziffern a_1 bis a_9, die das Buch identifizieren, nach der Regel

$$a_{10} \equiv -(a_1 + 2a_2 + \cdots + 9a_9) \bmod 11 \ ,$$

   wobei X für den Fall steht, das sich der Rest 10 ergibt.

   Später lässt sich dann überprüften, ob für die „gewichtete Quersumme“

$$s := \sum_{k=1}^{9} ka_k + a_{10}$$

   die Kongruenz $s \equiv 0 \bmod 11$ gilt. Wenn eine Ziffer falsch aufgenommen ist, erkennt man dies sofort: Hat man die Ziffer b anstelle von a_k, so ändert sich s um den Wert $kb - ka_k = k(b - a_k)$. Da 11 eine Primzahl ist, ist $k(b - a_k)$ nur dann durch 11 teilbar, wenn 11 bereits Teiler von $b - a_k$ ist, wenn also $b = a_k$ gilt.

   Aufgrund der unterschiedlichen Gewichte erkennt man auch einen anderen typischen Fehler, einen Zahlendreher. Hat man nämlich a_j und a_k vertauscht $(j \neq k)$, so ändert sich s um den Wert $ja_k + ka_j - ja_j - ka_k = (j - k)(a_k - a_j)$,

und dieser Wert ist wiederum nur dann kongruent 0 mod 11, wenn $a_j = a_k$ gilt, also gar kein echter Zahlendreher vorliegt.

Seit 2007 besteht die ISBN eines Buches aus 13 Ziffern $a_1, \ldots, a_{13}$, die letzte ist die Prüfziffer. Nun lautet die Prüfregel (die auch bei Strichcodes angewendet wird)

$$s' := a_1 + 3a_2 + a_3 + 3a_4 + \cdots + 3a_{12} + a_{13} \equiv 0 \bmod 10 \ .$$

Dass 10 nicht mehr prim ist, macht sich bemerkbar. Einzelfehler werden wieder sicher erkannt (warum?). Zahlendreher können aber schon für benachbarte Zahlen unbemerkt bleiben, nämlich genau dann, wenn $a_{i+1} = a_i \pm 5$ gilt. Dann ändert sich der Wert von s' um $a_i + 3a_{i+1} - 3a_i - a_{i+1} = 2(a_{i+1} - a_i) = \pm 10$ (im Fall, dass i geradzahlig ist, der andere Fall ist analog), und der Fehler tritt modulo 10 nicht mehr in Erscheinung.

5. **IBAN-Prüfziffern.** Die IBAN (international bank account number) eines Bankkontos ist nach den gleichen Schema konstruiert. Für Deutschland beginnt sie mit zwei Buchstaben und zwanzig folgenden Ziffern, die mit zwei Prüfziffern beginnen. Zur Berechnung der Prüfziffern werden beide Buchstaben durch jeweils zwei Ziffern ersetzt (DE = 1314) und zusammen mit den Prüfziffern ans Ende gestellt. Der entstehende 24-stellige Ausdruck $a_1 \ldots a_{24}$ wird als Dezimalzahl behandelt und modulo 97 betrachtet. Das Paar der Prüfziffern errechnen sich nach der Formel

$$10a_{23} + a_{24} \equiv -\sum_{i=1}^{22} a_i 10^{24-i} \bmod 97.$$

Schließlich werden die letzten 6 Ziffern wieder an den Anfang gestellt und die ersten 4 Ziffern in Buchstaben zurückverwandelt.

Die Rechnung modulo 97 mit solch großen Zahlen ist (anders als oft im Internet behauptet) gar nicht schwierig, wenn man die Zehnerpotenzen durch ihre Reste modulo 97 ersetzt. Man erhält die gewichtete Quersumme

$$\begin{aligned} 10a_{23} + a_{24} \equiv & -56a_1 - 25a_2 - 51a_3 - 73a_4 - 17a_5 - 89a_6 - 38a_7 - 62a_8 \\ & - 45a_9 - 53a_{10} - 15a_{11} - 50a_{12} - 5a_{13} - 49a_{14} - 34a_{15} \\ & - 81a_{16} - 76a_{17} - 27a_{18} - 90a_{19} - 9a_{20} - 30a_{21} - 3a_{22} \mod 97 \end{aligned}$$

Die Eigenschaften sind so gut wie bei der 10-stelligen ISBN: Weil 97 eine Primzahl ist, wird jeder Einzelfehler aufgedeckt. Weil die Gewichte alle verschieden sind, wird jeder Zahlendreher (benachbart oder unbenachbart) erkannt. Auch eine Verschiebung zweier (oder auch mehrerer) benachbarter Zahlen a_i, a_{i+1} um einen festen Wert b auf $a_i + b, a_{i+1} + b$, wie das z. B. bei Datenerfassung geschehen kann, bleibt nicht unbemerkt (warum?).

6. Modulares Rechnen wird für den Computer wichtig, wenn mit sehr großen Zahlen operiert wird, die auch vom Computer nicht mehr ohne weiteres verarbeitbar sind. Man wählt dann verschiedene Moduln $m_1, \cdots, m_r$ und rechnet

parallel mit ihnen. Am Ende muss man die verschiedenen Reste wieder zu einer Zahl zusammensetzen. Das geht ohne Schwierigkeiten, wenn $m_1, \ldots, m_r$ paarweise teilerfremd sind. Dann bestimmt sich aus den Restklassen modulo $m_1, \ldots, m_r$ eine eindeutige Restklasse modulo $m = m_1 \cdots m_r$, in der das Rechenresultat liegt („chinesischer Restsatz“), und sie legt die Zahl eindeutig fest, wenn nur m groß genug ist. – Zur Illustration multiplizieren wir die Zahlen 1010 und 997. Dazu setzen wir $m_1 = 99, m_2 = 100, m_3 = 101$. Es gilt $1010 \cdot 997 \equiv 20 \cdot 7 \equiv 41 \bmod 99$, $1010 \cdot 997 \bmod 10 \cdot (-3) \equiv -30 \bmod 100$, $1010 \cdot 997 \equiv 0 \cdot (-13) = 0 \bmod 101$. Der chinesische Restsatz ergibt $1010 \cdot 997 \equiv 7070 \bmod 999900$. Also $1010 \cdot 997 = 7070 + 999900 = 1006970$.

Beim Rechnen modulo m hat man es nicht mehr mit unendlich vielen ganzen Zahlen zu tun, sondern nur noch mit den m möglichen Resten $0, 1, \ldots, m-1$. Jeder solche Rest r steht stellvertretend für alle Zahlen a, die bei Division durch m genau den Rest r übriglassen. Oft fasst diese Zahlen zu einer „Restklasse“ zusammen, zum Beispiel bilden modulo 5 die Zahlen $\ldots, -8, -3, 2, 7, 12, \ldots$ eine Restklasse, die stellvertretend durch den Rest 2 repräsentiert wird.

Man kann sich also beim Rechnen modulo m vollständig auf die Zahlen $0, 1, \ldots, m-1$, die verschiedenen möglichen Reste, zurückziehen. Man schreibt dann $\mathbb{Z}_m = \{0, 1, \ldots, m-1\}$, wobei die Elemente von $\mathbb{Z}_m$ nicht mehr als normale ganze Zahlen aufzufassen sind, sondern als Restklassen. Dies ergibt neuartige Additions- und Multiplikationstabellen. Den Fall $m = 2$ (Rest 0 oder 1, gerade oder ungerade Zahlen) haben wir schon kennengelernt. Im Fall $m = 5$ sieht das so aus:

$+$	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	0
2	2	3	4	0	1
3	3	4	0	1	2
4	4	0	1	2	3

$\cdot$	0	1	2	3	4
0	0	0	0	0	0
1	0	1	2	3	4
2	0	2	4	1	3
3	0	3	1	4	2
4	0	4	3	2	1

In den Multiplikationstabellen steht rechts unten immer eine 1, wegen $(m-1)^2 = m^2 - 2m + 1 \equiv 1 \bmod m$. (Und was steht rechts unten in den Additionstabellen?)

In ein paar Punkten unterscheidet sich das Rechnen modulo m jedoch vom gewöhnlichen Rechnen mit ganzen Zahlen. Dies macht modulares Rechnen gerade auch für Anwendungen interessant.

Im Gegensatz zu den ganzen Zahlen kann das Produkt zweier Zahlen a, b modulo m als Resultat 0 ergeben, selbst wenn a und b nicht modulo m verschwinden. Etwa gilt $4 \cdot 7 \equiv 0 \bmod 14$, aber $4, 7 \not\equiv 0 \bmod 14$. Eine ganze Zahl a heißt *Nullteiler* modulo m, wenn $a \not\equiv 0 \bmod m$ gilt und es eine Zahl $b \not\equiv 0 \bmod m$ gibt mit

$$ab \equiv 0 \bmod m \ .$$

Im Zusammenhang damit steht, dass man aus modularen Gleichungen Faktoren nicht ohne weiteres wegkürzen darf. Zum Beispiel gilt

$$1 \cdot 2 \equiv 8 \cdot 2 \bmod 14 \quad \text{aber} \quad 1 \not\equiv 8 \bmod 14 \ .$$

Auch können modulo m zwei ganze Zahlen modulo m im Produkt 1 ergeben, auch wenn beide modulo m inkongruent ± 1 sind. Zum Beispiel gilt $3 \cdot 5 \equiv 1 \bmod 14$, aber $3, 5 \not\equiv \pm 1 \bmod m$. Eine ganze Zahl a heißt *invertierbar* modulo m, wenn es eine ganze Zahl c gibt, so dass

$$ac \equiv 1 \bmod m$$

gilt. c nennt man dann *Inverse* von a modulo m. Sie ist modulo m eindeutig bestimmt. Gilt nämlich auch $ac' \equiv 1 \bmod m$, so folgt

$$c' \equiv c'(ac) = c(ac') \equiv c \bmod m \ .$$

a heißt *Einheit* modulo m, wenn sogar $a^2 \equiv 1 \mod m$ gilt, a also seine eigene Inverse ist. So gilt $5^2 \equiv 1 \mod 24$.

Eine Zahl a kann modulo m nicht eine Inverse c besitzen und gleichzeitig Nullteiler sein. Denn gäbe es dann eine Zahl $b \not\equiv 0 \bmod m$ mit $ab \equiv 0 \bmod m$, so folgt $b = b \cdot 1 \equiv bac \equiv 0 \cdot c = 0 \bmod m$, ein Widerspruch.

Beispiel. Die Nullteiler mod15 sind die Zahlen 3,5,6,9,10,12, denn

$$3 \cdot 5 \equiv 6 \cdot 5 \equiv 9 \cdot 5 \equiv 3 \cdot 10 \equiv 12 \cdot 5 \equiv 0 \bmod 15 \ .$$

Die invertierbaren Zahlen mod15 sind 1,2,4,7,8,11,13,14, wie aus

$$1 \cdot 1 \equiv 2 \cdot 8 \equiv 4 \cdot 4 \equiv 7 \cdot 13 \equiv 11 \cdot 11 \equiv 14 \cdot 14 \equiv 1 \bmod 15 \ .$$

folgt. Es gibt 4 Einheiten modulo 15, nämlich 1, 4, 11, 14. □

Es ist nicht besonders schwer zu zeigen, dass jede ganze Zahl $a \not\equiv 0 \bmod m$ entweder Nullteiler oder invertierbar modulo m ist, und invertierbar genau dann, wenn a und m relativ prim sind. Im Fall einer Primzahl p haben alle Zahlen $a \not\equiv 0 \bmod p$ ein Inverses. Dies zeigt auch der folgende Satz (der „kleine Fermat“).

Satz von Fermat. *Ist p eine Primzahl und $a \not\equiv 0 \bmod p$, so folgt*

$$a^{p-1} \equiv 1 \bmod p \ .$$

Insbesondere ist a^{p-2} die Inverse von a.

Beweis. Wir stellen zunächst fest, dass für $0 < k < p$ der Binomialkoeffizient

$$\binom{p}{k} = \frac{p!}{k!(p-k)!}$$

ein Vielfaches von p ist, also modulo p verschwindet, denn der Faktor p im Zähler kann als Primzahl gegen keinen der Faktoren im Nenner weggekürzt werden, die alle kleiner als p sind.

Es folgt für ganze Zahlen $1 \leq a, b < p$ nach dem Binomischen Lehrsatz

$$(a+b)^p = \sum_{k=0}^{p} \binom{k}{p} a^k b^{p-k} \equiv a^p + b^p \bmod p$$

und damit

$$a^p \equiv (a-1)^p + 1 \equiv (a-2)^p + 2 \equiv \cdots \equiv 1^p + (a-1) \equiv a \bmod p \ .$$

Dies bedeutet, dass $a(a^{p-1} - 1) = a^p - a$ durch p teilbar ist. Da p nicht a teilt, muss es also als Primzahl $a^{p-1} - 1$ teilen. Das ist die Behauptung. □

Beispiel: Der Satz von Wilson. Sei p eine Primzahl. Wir bestimmen alle Einheiten modulo p: Aus $a^2 \equiv 1 \bmod p$ folgt, dass p die Zahl $a^2 - 1 = (a+1)(a-1)$ teilt. Da p Primzahl ist, muss p entweder $a-1$ oder $a+1$ teilen. Anders ausgedrückt bedeutet dies, dass $a \equiv 1 \bmod p$ oder $a \equiv -1 \bmod p$ gilt.

Wir haben also modulo p genau zwei Einheiten a zwischen 0 und p, nämlich 1 und $p-1$. Alle anderen Zahlen a besitzen ein Inverses ungleich a. Sie bilden unterschiedliche Paare, die jeweils im Produkt kongruent 1 modulo p sind. Bilden wir weiter das Produkt über alle diese Paare, so tritt jede Zahl $1 \leq a < p-1$ genau einmal als Faktor auf, und wir erhalten

$$(p-2)! \equiv 1 \bmod p \ .$$

Berücksichtigen wir noch die Gleichung $p - 1 \equiv -1 \bmod p$, so erhalten wir

$$(p-1)! \equiv -1 \bmod p \ .$$

Diese Kongruenz heißt Satz von Wilson. □

3.3 Öffentliche Chiffriersysteme

Die moderne geheime Nachrichtenübertragung beruht darauf, dass es Rechenaufgaben gibt, die ohne Hilfestellung nicht bewältigt werden können, die aber mit einer Zusatzinformation leicht zu handhaben sind. Diese Rechnungen finden nicht in den von der Schule her bekannten Zahlsystemen statt, man muss in neue Bereiche vorstoßen.

Die **Kryptographie** ist die Lehre von den Chiffriersystemen. Stellen wir uns vor, dass eine Person A eine geheime Nachricht an die Person B übermitteln möchte. A kodiert sie deswegen mittels einer bijektiven Kodierabbildung

$$\kappa : \mathcal{N} \to \mathcal{K}.$$

Anstelle der Nachricht $a \in \mathcal{N}$ im Klartext sendet A die chiffrierte Nachricht $\kappa(a)$. Der Empfänger dekodiert mittels der inversen Abbildung

$$\kappa^{-1} : \mathcal{K} \to \mathcal{N}.$$

Ein bekanntes Verfahren beruht auf der Addition modulo 2. Wir nehmen an, dass die Nachrichten als 01-Folgen der Länge n vorliegen, wählen also $\mathcal{N} = \mathcal{K} = \{0,1\}^n$. Wir fassen die Folgen als Vektoren der Läne n über dem Körper $\mathbb{Z}_2$ auf. Zum Kodieren wird ein 01-String $s = s_1 s_2 \dots s_n$ verwendet. Wir setzen

$$\kappa(a) \ := \ a + s,$$

die beiden 01-Folgen a und s werden also komponentenweise modulo 2 addiert. Der Schlüssel ist hier so lang wie die Nachricht. Es gilt $\kappa^{-1} = \kappa$, denn $s + s$ ist der nur aus Nullen bestehende Vektor. Nachrichten werden daher nach demselben Verfahren kodiert und dekodiert. Dieses klassische Verfahren hat den Namen ‚One-time-pad' (Einmalverschlüsselung). Es gilt $a + \kappa(a) = s$, gelingt es daher, eine kodierte Nachricht $k(a)$ zu entschlüsseln, so kennt man s und damit bereits die vollständige Kodier- und Dekodiervorschrift. Dies bedeutet, dass das Verfahren sicher ist, man hat keine Chance, eine Nachricht zu dechiffrieren, wenn man auf s keinen Zugriff hat. Ist s rein zufällig aus $\{0,1\}^n$ gewählt, so gilt dies auch für $\kappa(a)$. Die Methode ist perfekt, wenn s nur einmal verwendet wird.

Das RSA-Schema

Für klassische Chiffrierverfahren besteht ein Sicherheitsproblem darin, dass man nicht nur die Dekodiervorschrift κ^{-1} geheimhalten muß, sondern auch das Kodierverfahren κ. Wie das Beispiel des One-time-pad zeigt, lassen sich die Nachrichten mit Hilfe der Chiffrier- wie der Dechiffriervorschrift leicht entschlüsseln. Diffie und Hellman haben daher 1976 einen (damals beherzten) Vorschlag gemacht: Man solle Kodierabbildungen κ benutzen, für die $b = \kappa(a)$ aus a leicht berechnet werden kann, umgekehrt aber die Berechnung von $a = \kappa^{-1}(b)$ aus b typischerweise mit einem immensen Rechenaufwand verbunden ist, der praktisch nicht zu bewältigen ist (selbst wenn einem die Abbildung κ bekannt ist!). In dieser asymmetrischen Situation darf man das Chiffrierverfahren öffentlich bekannt machen, ohne die Sicherheit zu gefährden. Dies ist die Grundidee der modernen, computergestützten *öffentlichen Kodiersysteme*. Es bereitet Mühe, theoretisch zu begründen, dass solche Abbildungen κ, man spricht von *Einweg-(one-way-)Abbildungen*, wirklich existieren. Für praktische Zwecke haben sich jedoch verschiedene Abbildungen als

brauchbar erwiesen. Wir werden Abbildungen betrachten, bei denen das Dekodieren erst dann praktisch durchführbar ist, wenn man über einen zusätzlichen ‚Schlüssel' verfügt. Dann spricht man von *Falltür-(trapdoor-)Abbildungen.*

Zum Beispiel kann man schnell modulo m potenzieren:

$$a^{20} = a^{16+4} = (((a^2)^2)^2)^2 \cdot (a^2)^2.$$

Allgemein berechnet man a^c modulo m, indem man den Exponenten als binäre Zahl darstellt, $c = d_0 + d_1 \cdot 2 + \cdots + d_k \cdot 2^k$ mit $d_i \in \{0, 1\}$, in k Schritten rekursiv die Potenzen $a^{2^i} = (a^{2^{i-1}})^2$ modulo m berechnet und schließlich diejenigen Potenzen modulo m multipliziert, für die $d_i = 1$ gilt. Die Anzahl der Operationen ist von der Ordnung $O(k) = O(\log c)$. – Eine allgemeine Methode, mit der man schnell (wesentlich schneller als Durchprobieren) die Gleichung $b \equiv a^x \bmod m$ nach x auflöst (‚diskreter Logarithmus'), kennt man dagegen nicht.

Wir beschreiben nun das bekannteste öffentliche Chiffriersystem, das von Rivest, Shamir und Adleman 1978 vorgeschlagene *RSA-System.* Es baut darauf auf, dass es schwer ist, eine Zahl m in ihre Primfaktoren zu zerlegen.

Die RSA-Codierung.

- *Als Nachrichtenmenge $\mathcal{N} = \mathcal{K}$ wird $\mathbb{Z}_m$ gewählt. Dabei sei $m = pq$ das Produkt zweier sehr großer Primzahlen p und q. Wir gehen also davon aus, dass die Nachricht als natürliche Zahl $a < m$ dargestellt ist.*

- *Kodieren und Dekodieren geschieht mit natürlichen Zahl $s, t < (p-1)(q-1)$ mit der Eigenschaft*

$$s \cdot t \equiv 1 \bmod (p-1)(q-1) \ .$$

- *Die Kodierung der Nachricht $a \in \mathcal{N}$ ist dasjenige $b \in \mathcal{N}$ mit*

$$b \equiv a^s \bmod m \ .$$

  Die Dekodierung von $b \in \mathcal{N}$ ist dasjenige $c \in \mathcal{N}$ mit

$$c \equiv b^t \bmod m \ .$$

Beispiel. Wir wählen $m = 17 \cdot 23 = 391$, $s = 101$ und $t = 237$. Dann ist $st = 23937 \equiv 1 \bmod 352$. Die Nachricht $a = 52$ wird verschlüsselt zu

$$\begin{aligned} 52^{101} &= 52^{1+4+32+64} \\ &= 52 \cdot ((52)^2)^2 \cdot (((((52)^2)^2)^2)^2)^2 \cdot ((((((52)^2)^2)^2)^2)^2)^2 \\ &\equiv 52 \cdot 307 \cdot 188 \cdot 154 \equiv 358 \bmod 391 \end{aligned}$$

und wieder entschlüsselt als

$$\begin{aligned}358^{237} &= 358^{1+4+8+32+64+128}\\ &= 358\cdot((358)^2)^2\cdot(((358)^2)^2)^2\cdot(((((358)^2)^2)^2)^2)^2\\ &\quad\cdot((((((358)^2)^2)^2)^2)^2)^2\cdot(((((((358)^2)^2)^2)^2)^2)^2)^2\\ &\equiv 358\cdot 18\cdot 324\cdot 154\cdot 256\cdot 239\equiv 52 \bmod 391\ .\end{aligned}$$

Ohne Kenntnis von t müsste man alle Zahlen $a < 391$ durchgehen, ihre 101-te Potenz modulo 391 bilden, um diejenige zu finden, die dann 358 ergibt. □

Dieses Verfahren wird auf Computern implementiert mit Moduln m, die mehr als 200-stellig sind (Stand 2008). Man darf dieses Verfahren so einrichten, dass die Nachrichtenmenge $\mathcal{N}$, also m, die Zahl s und auch die kodierte Nachricht $\kappa(a)$ öffentlich zugänglich sind.

Es stellen sich einige Fragen. Wie findet man die Schlüssel? Warum ist es für die Sicherheit ausreichend, dass man nur p, q, und t geheim hält? Wir wollen hier nur zeigen, dass die Entschlüsselung korrekt funktioniert.

Satz. *Seien p, q Primzahlen, $m = pq$ und $st \equiv 1 \bmod (p-1)(q-1)$. Dann gilt*

$$b \equiv a^s \bmod m \quad \Leftrightarrow \quad a \equiv b^t \bmod m\ .$$

Beweis. Zu zeigen ist

$$(a^s)^t \equiv a \bmod m\ .$$

Sei zunächst $a \not\equiv 0 \bmod p$. Dann folgt $a^{p-1} \equiv 1 \bmod p$ nach dem Satz von Fermat. Nach Voraussetzung gibt es eine natürliche Zahl k, so dass $st = 1 + k(p-1)(q-1)$. Es folgt

$$(a^s)^t = a\cdot(a^{p-1})^{k(q-1)} \equiv a \bmod p\ .$$

Offenbar gilt diese Aussage auch für $a \equiv 0 \bmod p$. Genauso folgt die Kongruenz $(a^s)^t \equiv a \bmod q$ für alle a. Insgesamt ist $(a^s)^t - a$ durch p und q teilbar, also auch durch m. Dies ergibt die Behauptung. □

Das RSA-System steht und fällt damit, dass die Primfaktoren p und q geheim bleiben, sonst kann man aus s leicht t bestimmen und die Geheimbotschaft damit knacken. Die Erfahrung zeigt, dass die Faktorisierung von m in seine Primteiler einen hohen Rechenaufwand erfordert und praktisch nicht mehr gelingt, wenn p und q ausreichend groß sind. Dies ist das *Grundpostulat der Kryptographie*: Man geht davon aus, dass es keinen schnellen Algorithmus zum Zerlegen einer Zahl in seine Primfaktoren gibt. Man nennt deshalb in der Kryptographie ein Chiffriersystem ‚beweisbar sicher', wenn es letztlich auf die Faktorisierung großer Zahlen zurückgeführt werden kann.

Signatur von Nachrichten. Das RSA-Schema eignet sich gut zum geheimen Austausch von Nachrichten innerhalb einer Gruppe von Teilnehmern. Jeder Teilnehmer A erhält von einer Zentrale einen öffentlichen Schlüssel $(m(A), s(A))$ und seinen persönlichen geheimen Schlüssel $t(A)$. A teilt B eine geheime Nachricht a als

$$b := \kappa_B(a) \equiv a^{s(B)} \bmod m(B)$$

mit, und B dekodiert mittels

$$\kappa_B^{-1}(b) \equiv b^{t(B)} \bmod m(B) \ .$$

Ein vorheriger Kontakt zwischen A und B ist nicht erforderlich, A braucht lediglich den öffentlich zugänglichen Schlüssel $s(B)$ von B. Ein wichtiger Vorteil des Systems ist, dass es den Teilnehmern erlaubt, Mitteilungen fälschungssicher zu signieren (‚Unterschriftensystem'). A beglaubigt eine öffentliche Nachricht a durch die öffentliche Mitteilung von

$$c := \kappa_A^{-1}(a) \equiv a^{t(A)} \bmod m(A) \ .$$

Alle anderen Teilnehmer können verifizieren, dass $a \equiv c^{s(A)} \bmod m(A)$ gilt.

4 Zufall

4.1 Wahrscheinlichkeiten

Wahrscheinlichkeiten benutzt man dazu, um die Chancen zu quantifizieren, dass *Zufallsereignisse* eintreten. Wir schreiben

$$\mathbf{P}(E)$$

für die Wahrscheinlichkeit des Ereignisses E. Das *Komplementärereignis* (Gegenereignis) ist das Ereignis E^c, das genau dann eintritt, wenn E nicht eintritt. Es hat Wahrscheinlichkeit

$$\mathbf{P}(E^c) = 1 - \mathbf{P}(E) \ .$$

Die Frage, was das Wesen von Zufallsereignissen und Wahrscheinlichkeiten eigentlich ausmacht, lassen wir beiseite. Man ist sich nicht einig, ob es den Zufall in der Realität wirklich gibt oder ob er allein ein Gedankenkonstrukt ist. Über seine formalen Eigenschaften hat man sich aber verständigen können, und es bewährt es sich immer wieder, mit Wahrscheinlichkeiten zu arbeiten. Wenig stritt ist auch die Behandlung konkreter Beispiele.

Beispiel. Beim Werfen eines (ungezinkten) Würfels hat das Ereignis

$$E_6 = \{\text{eine 6 wird geworfen}\}$$

zweifellos die Wahrscheinlichkeit

$$\mathbf{P}(E_6) = \mathbf{P}(\text{eine 6 wird geworfen}) = \frac{1}{6}$$

und das Komplementärereignis

$$E_6^c = \{\text{es fällt eine der Zahlen } 1, \ldots, 5\}$$

die Wahrscheinlichkeit 5/6.

Beispiel. Reihenuntersuchungen. Bei der Wahrscheinlichkeit des Ereignisses

$$E_{kr} = \{\text{eine Person trägt die Krankheit K in sich}\}$$

denkt man an den relativen Anteil der Personen, die von dieser Krankheit K befallen sind – etwa unter allen 40–50-jährigen Personen, die aus einer bestimmten Region stammen – und die noch keine Symptome der Krankheit zeigen müssen. In relevanten Fällen gilt

$$\mathbf{P}(E_{kr}) = 0,008 = 0,8\%$$

Damit quantifiziert man die Chancen, dass ein ganz zufällig ausgewähltes Individuum aus der Personengruppe die Krankheit trägt. (Für Personen mit Krankheitssymptomen ist diese Wahrscheinlichkeit natürlich nicht anwendbar.) Das Gegenereignis

$$E_{ges} = \{\text{eine Person ist frei von K}\}$$

hat dann Wahrscheinlichkeit $0,992$. Wir werden sehen, was solche Zahlen im Kontext von Reihenuntersuchungen bedeuten. (Eine genauere Erörterung dieser Thematik findet sich in dem empfehlenswerten Buch ‚Das Einmaleins der Skepsis' von G. Gigerenzer.)

Neben Wahrscheinlichkeiten betrachten wir bedingte Wahrscheinlichkeiten. Die *bedingte Wahrscheinlichkeit von E_1 gegeben E_2*,

$$\mathbf{P}(E_1 \,|\, E_2) \ ,$$

ist anschaulich gesprochen die Wahrscheinlichkeit des Ereignisses E_1, wenn schon bekannt ist, dass das Ereignis E_2 eingetreten ist.

Beispiel. Beim Würfeln gilt

$$\mathbf{P}(\text{eine 6 fällt} \,|\, \text{die geworfene Zahl ist gerade}) = 1/3 \ .$$

Denn wenn man schon weiß, dass die geworfene Zahl gerade ist, ändert sich die Chance für eine 6 offenbar von $1/6$ auf $1/3$.

Beispiel. Reihenuntersuchungen. Bedingte Wahrscheinlichkeiten werden gebraucht, um die Güte von Vorsorgeuntersuchungen („Screenings") für die Krankheit K zu bewerten. Mit hoher Wahrscheinlichkeit hat man einen positiven Befund, gegeben die Testperson ist krank, manchmal ist aber auch bei gesunden Personen der Befund positiv. Bezeichnet

$$E_{po} = \{\text{positiver Befund für die Testperson}\}$$

das Ereignis, das für eine Person der Testbefund positiv ausfällt, so sind typische Werte

$$\mathbf{P}(E_{po} \,|\, E_{kr}) = 0,9 = 90\% \ , \quad \mathbf{P}(E_{po} \,|\, E_{ges}) = 0,07 = 7\% \ .$$

Für eine Person mit positivem Testbefund sind diese bedingten Wahrscheinlichkeiten jedoch weniger interessant, sie möchte vor allem wissen, mit welcher Wahrscheinlichkeit sie krank ist, wie groß also die bedingte Wahrscheinlichkeit

$$\mathbf{P}(E_{kr} \,|\, E_{po})$$

ist. Wir werden sehen, wie sich diese Wahrscheinlichkeit berechnet und dass sie typischerweise kleiner ist, als man erwarten könnte.

Es geht also um den Zusammenhang zwischen zwei Ereignissen E_1 und E_2. Man kann ihn auf drei Weisen erfassen.

Szenario 1. Für das Ereignis, dass E_1 und E_2 gleichermaßen eintreten, schreibt man

$$E_1 \cap E_2 := \{E_1 \text{ und } E_2 \text{ treten beide ein}\} ,$$

genauso $E_1 \cap E_2^c$ für das Ereignis, dass E_1 eintritt und E_2 nicht eintritt usw. Prinzipiell ist dann durch die vier Wahrscheinlichkeiten des Eintretens bzw. Nichteintretens

$$p(ee) := \mathbf{P}(E_1 \cap E_2) \ , \ p(en) := \mathbf{P}(E_1 \cap E_2^c) \ ,$$
$$p(ne) := \mathbf{P}(E_1^c \cap E_2) \ , \ p(nn) := \mathbf{P}(E_1^c \cap E_2^c)$$

alles festgelegt. Ihre Summe ist 1, denn genau eines der vier Ereignisse tritt sicher ein. Es gilt

$$\mathbf{P}(E_1) = p(ee) + p(en) \ , \quad \mathbf{P}(E_2) = p(ee) + p(ne) \ .$$

Denn E_1 tritt ein, wenn $E_1 \cap E_2$ oder $E_1 \cap E_2^c$ eintritt, und diese beiden Ereignisse schließen sich gegenseitig aus.

Beispiel. Reihenuntersuchungen. Beim Gesundheitstest reagieren in der Population der Größe 10.000 von den 9.920 gesunden Personen der Anteil 698 positiv auf den Test, und von den 80 Kranken 72 positiv. Für $E_1 = E_{kr}$, $E_2 = E_{po}$ sind dann die Werte

$$p(ee) = \frac{72}{10.000} \ , \ p(en) = \frac{8}{10.000} \ , \ p(ne) = \frac{698}{10.000} \ , \ p(nn) = \frac{9.222}{10.000} \ .$$

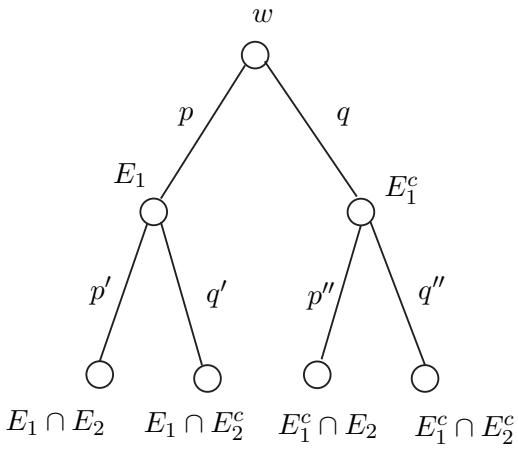
und

$$\mathbf{P}(E_{kr}) = \frac{80}{10.000} = 0,008 \ , \quad \mathbf{P}(E_{po}) = \frac{770}{10.000} = 0,077$$

Szenario 2. Oft ist der Ausgangspunkt ein anderer. Man stellt sich vor, dass sich erst herausstellt, ob E_1 eintritt, und dass, in Abhängigkeit davon, sich dann das Eintreten von E_2 entscheidet. In diesem Szenario sind statt $p(ee), \ldots, p(nn)$ die unbedingten und bedingten Wahrscheinlichkeiten

$$p := \mathbf{P}(E_1) \ , \quad p' := \mathbf{P}(E_2 \mid E_1) \ , \quad p'' := \mathbf{P}(E_2 \mid E_1^c)$$

vorgegeben. Der Entscheidungsverlauf über das Eintreten von E_1 und E_2 wird im folgenden Baum dargestellt:

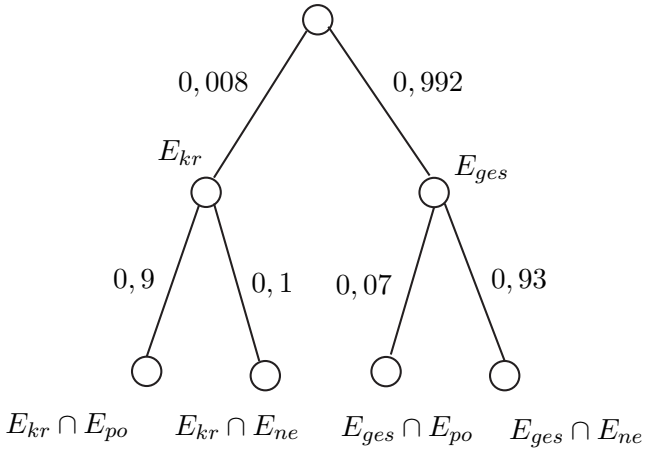


Ausgehend von der Wurzel w geht man mit Wahrscheinlichkeit p nach links unten und mit der Gegenwahrscheinlichkeit $q := 1 - p$ nach rechts unten. Im zweiten Schritt geht man dann mit Wahrscheinlichkeit p' bzw. p'' wieder nach links und mit den Gegenwahrscheinlichkeiten q' bzw. q'' nach rechts.

Beispiel. Reihenuntersuchungen. In diesem Fall, mit $E_{ges} := E_{kr}^c$, $E_{ne} = E_{pos}^c$, sind die Wahrscheinlichkeiten $p = \mathbf{P}(E_{kr}) = 0,008$ und

$$p' = \mathbf{P}(E_{po} \mid E_{kr}) = \frac{72}{80} = 0,9 \ , \quad p'' = \mathbf{P}(E_{po} \mid E_{ges}) = \frac{698}{9920} = 0,07 \ ,$$

und der Baum hat die Gestalt



Die Wahrscheinlichkeiten, in einem der vier Blättern zu landen, ergeben sich durch Multiplikation der Kantengewichte als

$$p(ee) = pp' \ , \ p(en) = pq' \ , \ p(ne) = qp'' \ , \ p(nn) = qq'' \ ,$$

und die Wahrscheinlichkeit, dass E_2 eintritt, ermittelt sich als

$$\mathbf{P}(E_2) = p(ee) + p(ne) = pp' + qp''$$

bzw. als

$$\mathbf{P}(E_2) = \mathbf{P}(E_2 \mid E_1)\mathbf{P}(E_1) + \mathbf{P}(E_2 \mid E_1^c)\mathbf{P}(E_1^c) \ . \tag{5}$$

Diese Formel heißt der *Satz von der totalen Wahrscheinlichkeit*. Im Beispiel bedeutet dies

$$\mathbf{P}(E_{po}) = 0,9 \cdot 0,008 + 0,07 \cdot 0,992 = 0,077 \ .$$

Umgekehrt bestimmen sich auch p, p', p'' aus den Wahrscheinlichkeiten $p(ee), \ldots, p(nn)$, gemäß

$$p = p(ee) + p(en) \ , \quad p' = \frac{p(ee)}{p} \ , \quad p'' = \frac{p(ne)}{1-p} \ .$$

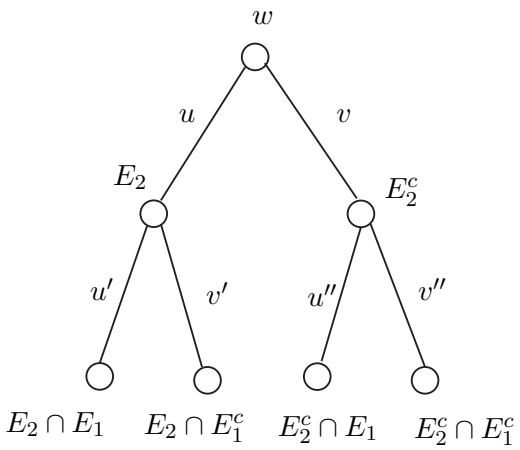
Die zweite Formel können wir auch schreiben als

$$\mathbf{P}(E_2 \,|E_1) = \frac{\mathbf{P}(E_1 \cap E_2)}{\mathbf{P}(E_1)} \ , \tag{6}$$

was der üblichen Definition von bedingten Wahrscheinlichkeiten entspricht.

Damit lassen sich beide Szenarien ineinander überführen, sie sind zwei Seiten ein und derselben Medaille.

Szenario 3. Wenn also alles durch $p(ee), \ldots, p(nn)$ bestimmt ist, dann können wir im zweiten Szenario auch die Rolle von E_1 und E_2 vertauschen und uns vorstellen, dass erst entschieden wird, ob E_2 eintritt, und dann in Abhängigkeit davon sich entscheidet, ob E_1 eintritt. Zu dem obigen Baum tritt ein äquivalenter Baum



Die neuen Kantenwahrscheinlichkeiten u, $v = 1 - u$, ... berechnen sich in analoger Weise aus $p(ee) \ldots, p(nn)$.

Die Äquivalenz der drei Szenarien bedeutet insbesondere, dass sich die Wahrscheinlichkeiten aus Szenario 2 in diejenigen aus Szenario 3 umrechnen lassen und insbesondere die bedingte Wahrscheinlichkeit $\mathbf{P}(E_1 \,|E_2)$ aus den Wahrscheinlichkeiten $\mathbf{P}(E_1)$, $\mathbf{P}(E_2 \,|E_1)$ und $\mathbf{P}(E_2 \,|E_1^c)$ berechnen lässt. Diese Erkenntnis war durchaus ein Aha-Erlebnis in der Geschichte der Wahrscheinlichkeitsrechnung. Die folgende Formel leistet die Umrechnung.

Satz. Die Formel von Bayes. *Es gilt*

$$\mathbf{P}(E_1 \,|\, E_2) = \frac{\mathbf{P}(E_2 \,|\, E_1)\mathbf{P}(E_1)}{\mathbf{P}(E_2 \,|\, E_1)\mathbf{P}(E_1) + \mathbf{P}(E_2 \,|\, E_1^c)\mathbf{P}(E_1^c)} \ .$$

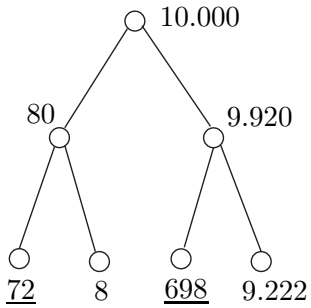
Beweis. Nach (6) gilt $\mathbf{P}(E_1 \,|\, E_2)\mathbf{P}(E_2) = \mathbf{P}(E_1 \cap E_2) = \mathbf{P}(E_2 \,|\, E_1)\mathbf{P}(E_1)$. Die Behauptung folgt dann per Division durch $\mathbf{P}(E_2)$ unter Beachtung der Formel (5). □

Beispiel. Reihenuntersuchungen. Mit $E_1 = E_{kr}$ und $E_2 = E_{po}$ ergibt sich

$$\mathbf{P}\big(\text{Person ist krank} \,|\, \text{positiver Befund}\big) = \frac{0,9 \cdot 0,008}{0,9 \cdot 0,008 + 0,07 \cdot 0,992} = 0,09 \ .$$

Dies ist bemerkenswert. Wir erkennen: Wenn man ein positives Testresultat hat, ist die Chance viel größer, einer von den vielen Gesunden zu sein, bei dem der Test einmal nicht richtig funktioniert, als einer der wenigen Kranken. Betroffenen mit positivem Testbefund kann man also nur raten, die Ruhe zu bewahren und sich erst einmal von einem Spezialisten gründlich untersuchen zu lassen.

Das Resultat wird plastisch, wenn wir den Baum aus Szenario 2 in den absoluten Häufigkeiten der 10.000 Testpersonen darstellen:



Die 770 Personen mit positivem Testresultat setzen sich aus 698 Gesunden und nur 72 Kranken zusammen, und es gilt (gerundet) $72/770 = 0,09$.

Das Simpson-Paradox. Mit dem Satz von der totalen Wahrscheinlichkeit kann man auch ein paradoxes Phänomen aufklären. Wir betrachten es erst im Beispiel.

Der folgende Datensatz betrifft die Wirkung zweier Methoden A und B zur Behandlung von Nierensteinen. In der Studie wurden einerseits an Patienten mit kleinen Nierensteinen verglichen, die beiden Testgruppen hatten die Größe $m_A = 87$ und $m_B = 270$, die Anzahl der erfolgreichen Behandlungen war hier $e_A = 81$ und $e_B = 234$. Die Erfolgswahrscheinlichkeiten der Methoden A und B berechnen sich als

$$\frac{e_A}{m_A} = \frac{81}{87} = 0,93 \ , \quad \frac{e_B}{m_B} = \frac{234}{270} = 0,87 \ .$$

Für zwei andere Gruppen von Patienten mit großen Nierensteinen ergaben sich die Werte

$$\frac{f_A}{n_A} = \frac{192}{263} = 0,73 \ , \quad \frac{f_B}{n_B} = \frac{55}{80} = 0,69 \ .$$

Methode A schneidet also sowohl bei kleinen wie bei großen Steinen besser ab. Fasst man jedoch die Gruppen zusammen und unterscheidet nicht mehr nach der Art der Steine, so kehrt sich der Sachverhalt um:

$$\frac{e_A + f_A}{m_A + n_A} = \frac{273}{350} = 0,78 \ , \quad \frac{e_B + f_B}{m_B + n_B} = \frac{289}{350} = 0,83 \ .$$

Nun scheint Methode B erfolgreicher, ein Trugschluss.

Kurz gesagt geschieht das Folgende: In der Studie wurde Methode A überwiegend auf Patienten mit großen Steinen angewandt, Methode B jedoch überwiegend auf Patienten mit kleinen Steinen. Generell sind die Erfolgsaussichten jedoch bei Patienten mit großen Steinen geringer, was sich in den Daten dann für die Methode A deutlich stärker bemerkbar macht als für die Methode B.

Um den Trugschluss genauer zu verstehen, stellen wir zunächst fest, dass man im Allgemeinen aus den *Werten* von Brüchen

$$\frac{e}{m} \text{ und } \frac{f}{n}$$

nicht auf den Wert von

$$\frac{e+f}{m+n}$$

(der „falschen Schülersumme“ der Brüche) schließen kann. Vielmehr braucht man noch Angaben über die Zähler und Nenner (die ja erst einmal nicht gekürzt zu sein brauchen). Es gilt nämlich

$$\frac{e+f}{m+n} = \frac{m}{m+n}\frac{e}{m} + \frac{n}{m+n}\frac{f}{n} = \lambda\frac{e}{m} + (1-\lambda)\frac{f}{n}$$

mit

$$\lambda := \frac{m}{m+n} = \frac{1}{1+\frac{n}{m}} \ .$$

Es ist also $\frac{e+f}{m+n}$ ein gewichtetes Mittel von $\frac{e}{m}$ und $\frac{f}{n}$. Für λ sind verschiedene Werte zwischen 0 und 1 möglich, für $\frac{e+f}{m+n}$ ergeben sich dann unterschiedliche Werte zwischen $\frac{e}{m}$ und $\frac{f}{n}$. Um auch λ festzulegen, braucht man noch Verhältnis n/m.

Beispiel. Wenn von zwei Flaschen die eine m Liter Flüssigkeit enthält, davon e Liter Fruchtsaft, die andere n Liter mit f Liter Fruchtsaft, so hat man nach Zusammenschütten $m+n$ Liter mit $e+f$ Litern Fruchsaft. Offenbar ist dann die Endkonzentration $\frac{e+f}{m+n}$ nicht nur von beiden Ausgangskonzentrationen $\frac{e}{m}$ und $\frac{f}{n}$ abhängig, sondern auch vom Verhältnis der Flüssigkeitsmengen, von $\frac{m}{n}$ bzw. von $\lambda = \frac{m}{m+n}$.

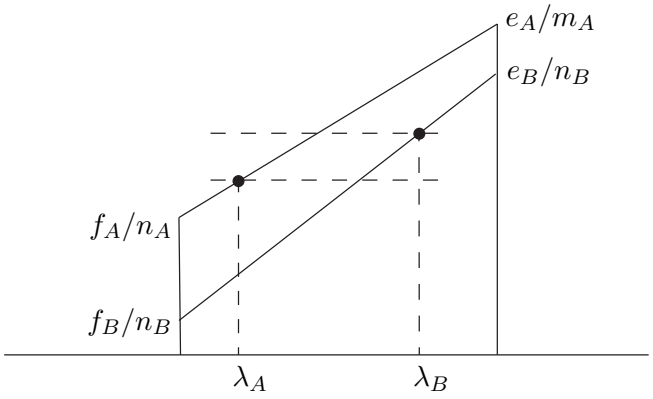
Beispiel. Das Torverhältnis einer Fußballmannschaft ist in der Hinrunde $\frac{a}{b}$ und in der Rückrunde $\frac{c}{d}$. Dann ist ihr Torverhältnis insgesamt $\frac{a+c}{b+d}$.

Aufgabe. In einem Dorf sind $\frac{2}{3}$ der erwachsenen Männer mit $\frac{5}{8}$ der erwachsenen Frauen verheiratet. Welcher Anteil aller Erwachsenen ist dann verheiratet?

Zurück zu den beiden Behandlungstechniken: Der entscheidende Punkt ist, dass in unserem Beispiel der Parameter λ für die Methoden A und B verschiedene Werte annimmt, nämlich

$$\lambda_A = \frac{m_A}{m_A + n_A} = \frac{87}{350}\ , \quad \lambda_B = \frac{m_B}{m_B + n_B} = \frac{270}{350}\ .$$

Dies führt zu der vermeintlichen Umkehrung, wie die folgende Graphik visualisiert.



Wir überführen unsere Überlegungen nun in die Sprache der Stochastik. Dazu seien E und E' zwei Ereignisse. Im Beispiel sind das

$$E = \{\text{Patient hat kleine Steine}\}$$
$$E^c = \{\text{Patient hat große Steine}\}$$
$$E' = \{\text{die Behandlung ist erfolgreich}\}$$

Es geht um den Sachverhalt, dass für zwei verschiedene Wahrscheinlichkeitsbelegungen $\mathbf{P}_A$ und $\mathbf{P}_B$ aus den Ungleichungen

$$\mathbf{P}_A(E' \,|\, E) \leq \mathbf{P}_B(E' \,|\, E) \ , \quad \mathbf{P}_A(E' \,|\, E^c) \leq \mathbf{P}_B(E' \,|\, E^c)$$

im Allgemeinen nicht auf $\mathbf{P}_A(E') \leq \mathbf{P}_B(E')$ geschlossen werden kann. Dem liegt zugrunde, dass auch hier sich eine Wahrscheinlichkeit $\mathbf{P}(E')$ als gewichtetes Mittel von $\mathbf{P}_A(E' \,|\, E)$ und $\mathbf{P}_B(E' \,|\, E)$ berechnet, gemäß dem Satz von der totalen Wahrscheinlichkeit

$$\mathbf{P}(E') = \mathbf{P}(E' \mid E)\mathbf{P}(E) + \mathbf{P}(E' \,|\, E^c)\mathbf{P}(E^c) \ .$$

Die Rolle des Faktors λ übernimmt die Wahrscheinlichkeit $\mathbf{P}(E)$.

4.2 Variabilität

Bei einer Meinungsumfrage möchte man feststellen, wie groß der relative Anteil p von Personen ist, die einer bestimmten Ansicht zustimmt („Ist Ihnen Freiheit wichtiger als Gerechtigkeit?“). Dazu nimmt man eine zufällige Stichprobe der Länge n und betrachtet

$$X := \text{ Anzahl der Ja-Antworten in der Stichprobe .}$$

X ist eine Zufallsvariable, wie ihr Wert ausfällt, hängt von der Wahl der Stichprobe ab.

Es liegt auf der Hand, wie man dann p schätzt, nämlich durch

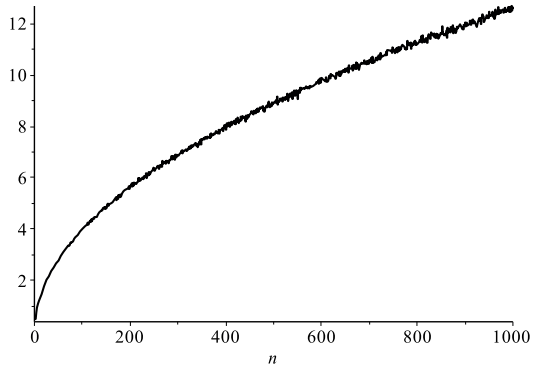
$$\widehat{p} := \frac{X}{n} \ ,$$

der relativen Häufigkeit der Ja-Antworten. Dagegen ist gar nicht offensichtlich, wie groß man n zu wählen hat, um die zufällige Variabilität von $\widehat{p}$ ausreichend klein zu bekommen. Es stellt sich etwa die Frage: In welchem Maß muss man n vergrößern, um die die Variabilität zu halbieren? Wir werden sehen, dass es dazu nicht langt, n zu verdoppeln.

Um die Variabilität von X aufzuspüren, kann man ein Computerexperiment machen: Man erzeugt mit dem Rechner eine Anzahl künstlicher Stichproben $X_{1n}, \ldots, X_{rn}$, sagen wir $r = 10.000$ Stück, von der Länge n und zur Erfolgswahrscheinlichkeit p (oder, wenn p nicht gegeben ist, zur beobachteten Erfolgswahrscheinlichkeit $\widehat{p}$). Die Variabilität in den X_{in} bei festem n kann man dann durch ihre mittlere Abweichung von np ausdrücken, also durch

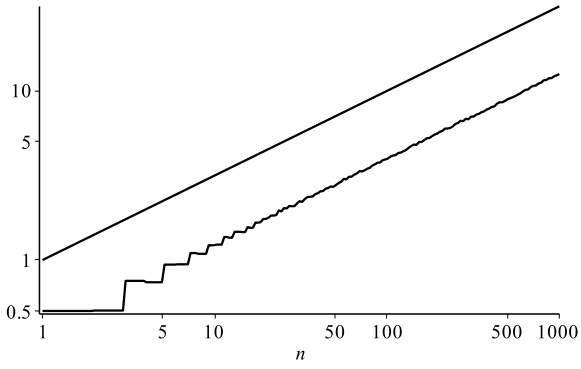
$$Y_n := \frac{1}{r} \sum_{i=1}^{r} \big| X_{in} - np \big| \ .$$

Wie ein Computersimulation mit $p = \frac{1}{2}$ zeigt, wächst diese Größe nicht linear in n.



Die Vermutung liegt nahe, dass Y_n stattdessen proportional zu $\sqrt{n}$ wächst. Dies lässt sich am besten in doppelt-logarithmischer Skala untersuchen: Die Gleichung $y = cx^a$ geht durch Logarithmieren in $\log y = a \log x + \log c$ über, und der Graph der Funktion wird, wenn man $\log y$ gegen $\log x$ abträgt (statt y gegen x), zu einer Geraden, deren Steigung für eine Wurzelfunktion gleich $a = \frac{1}{2}$ ist.

Die Simulationsdaten bestätigen die Vermutung für ausreichend großes n völlig (die obere Gerade stellt $y = \sqrt{x}$ dar)



Die Variabilität von $\widehat{p}$ lässt sich analog aus

$$\widehat{p}_{1n} = \frac{X_{1n}}{n}, \ldots, \widehat{p}_{rn} = \frac{X_{rn}}{n}$$

erschließen, bzw.

$$\frac{Y_n}{n} = \frac{1}{r} \sum_{i=1}^{r} |\widehat{p}_{in} - p| \ ,$$

der mittleren Abweichung der $\widehat{p}_{in}$ von p. Diese Größe ist nun proportional zu $n^{-1/2}$. Dies sind die $\sqrt{n}$-Regeln der Stochastik. Wir erkennen: Um den Schätzfehler von $\widehat{p}$ zu halbieren, müssen wir die Stichprobengröße n vervierfachen.

Ein vertieftes Verständnis entsteht, wenn man die Zufallsvariable von einem theoretischen Standpunkt betrachtet. Es ist nicht schwer, die Wahrscheinlichkeit

$$\mathbf{P}(X = k) \ ,$$

dass X den Wert k annimmt, auszurechnen ($k = 0, 1, \ldots, n$). Aber das brauchen wir hier gar nicht. Es geht uns nur um den mittleren Wert und die Variabilität von X.

Dazu berechnet man erstens den *Erwartungswert* von X. Er ist das gewichtete Mittel der möglichen Werte $k = 0, 1, \ldots, n$ von X, gewichtet mit ihren Eintrittswahrscheinlichkeiten,

$$\mathbf{E}[X] := \sum_{k=0}^{n} k\mathbf{P}(X = k) \ .$$

Wir werden später zeigen, dass sich

$$\mathbf{E}[X] = np$$

ergibt, ein Resultat, das einem auch der gesunde Menschenverstand sagt. Mittels Division durch n folgt

$$\mathbf{E}[\widehat{p}] = \frac{\mathbf{E}[X]}{n} = p \ .$$

Man sagt, $\widehat{p}$ ist ein erwartungstreuer Schätzer von p. Im Mittel gibt also $\widehat{p}$ den richtigen Wert.

Um zweitens auch ein Maß für die Variabilität von X zu erhalten, wäre es wünschenswert, eine ähnlich einfache Formel für den Erwartungswert

$$\mathbf{E}\big[|X - np|\big] = \sum_{k=0}^{n} |k - np|\mathbf{P}(X = k)$$

zu erhalten. Exakt ist dies nicht gelungen, und approximativ auch nur mit einem gewissen Aufwand. Deswegen betrachtet man ersatzweise die *Varianz* von X, den mittleren *quadratischen* Abstand zwischen X und np,

$$\mathbf{Var}[X] := \mathbf{E}\big[(X - \mathbf{E}[X])^2\big] = \sum_{k=0}^{n} (k - np)^2\mathbf{P}(X = k) \ .$$

Diese Größe erschließt sich rechnerisch viel besser, nicht nur in unserem Zusammenhang. Wir werden sehen, dass sich

$$\mathbf{Var}[X] = npq \ , \quad \text{mit } q := 1 - p$$

ergibt.

Die Varianz ist noch ungeeignet als Kennzahl für die Variabilität von X, die Wirkung des Quadrierens muss man erst wieder rückgängig machen. Dies führt zur Wurzel der Varianz

$$\sqrt{\mathbf{Var}[X]} = \sqrt{npq} \ ,$$

sie ist gut geeignet. Sie heißt die *Standardabweichung* oder *Streuung* von X. Bemerkenswert ist, dass sie mit $\sqrt{n}$ wächst, ähnlich, wie sich dies schon im Computerexperiment gezeigt hat.

Daraus ergibt sich weiter, dass die Standardabweichung von $\widehat{p}$ gleich

$$\sqrt{\mathbf{Var}[\widehat{p}]} = \frac{\sqrt{\mathbf{Var}[X]}}{n} = \frac{\sqrt{pq}}{\sqrt{n}}$$

ist. Wir sehen erneut: Um die Genauigkeit der Schätzung zu verdoppeln, muss man n vervierfachen!

Beispiel. Verstecken hinter dem Zufall. Bei heiklen Themen („Waren Sie schon untreu?“) kann man den relativen Anteil π der schwarzen Schafe nicht durch direktes Befragen ermitteln, die Verfälschung ist unkontrollierbar. Hier kann man sich wie folgt aushelfen: Die Testperson wirft zunächst eine gezinkte Münze, die Kopf und Wappen mit Wahrscheinlichkeit ρ bzw. $1-\rho$ zeigt. Bei Kopf soll die Frage richtig mit Ja oder Nein beantwortet werden, bei Wappen aber falsch, die Person soll Nein statt Ja und Ja statt Nein antworten. Hat die Testperson die Münze heimlich geworfen, so ist ihr mit ihrer Antwort nicht mehr auf die Schliche zu kommen, bei $\rho = 1/3$ etwa kann sie sich sicher fühlen (was ist mit $\rho = 1/2$?). Die Wahrscheinlichkeit für eine Ja-Antwort ist dann

$$p = \pi\rho + (1-\pi)(1-\rho) \ .$$

Für $\rho \neq \frac{1}{2}$ folgt

$$\pi = \frac{p+\rho-1}{2\rho-1} \ ,$$

ein natürlicher Schätzer für π ist daher

$$\widehat{\pi} = \frac{X/n+\rho-1}{2\rho-1} = \frac{X}{n(2\rho-1)} + \frac{\rho-1}{2\rho-1} \ .$$

Ihre Streuung ist dieselbe wie die von $X/n(2\rho-1)$. Nun zeigt ein Rechnung, dass

$$p(1-p) = \rho(1-\rho) + \pi(1-\pi)(2\rho-1)^2 \ ,$$

und für die Streuung von $\widehat{\pi}$ ergibt sich

$$\frac{\sqrt{np(1-p)}}{n(2\rho-1)} = \frac{1}{\sqrt{n}}\sqrt{\frac{\rho(1-\rho)}{(2\rho-1)^2} + \pi(1-\pi)} \ .$$

Der Preis also, der für den Schutz der Befragten (und damit eine verlässliche Auswertung) zu zahlen ist, ist eine kräftig vergrößerte Streuung. Der Faktor liegt für $\rho = 1/3$ zwischen $\sqrt{2} = 1,41$ und $\sqrt{2+1/4} = 1,5$. Man kann sich nun überlegen, wie hoch n zu wählen ist, um eine gewisse Genauigkeit zu erhalten.

Berechnung von Erwartungswert und Varianz. Man kann für die Zufallsvariable X die Wahrscheinlichkeiten $\mathbf{P}(X = k)$ und daraus den Erwartungswert und die Varianz direkt berechnen. Aufschlussreicher ist ein anderer Weg.

Wir arbeiten mit der Zerlegung

$$X = Z_1 + \cdots + Z_n$$

mit

$$Z_i := \begin{cases} 1\ , & \text{falls die } i\text{-te Person mit „Ja“ antwortet ,} \\ 0 & \text{andernfalls .} \end{cases}$$

Für jede Ja-Antwort wird also eine 1 gezählt, und das gibt in der Summe die X Ja-Antworten.

Nun gilt eine fundamentale Eigenschaft des Erwartungswertes, seine Linearität. Sie lautet

$$\mathbf{E}[cX + dY] = c\mathbf{E}[X] + d\mathbf{E}[Y]$$

für zwei reellwertige Zufallsvariable X, Y und reelle Zahlen c, d. Der Beweis ist nicht schwer zu führen (zumindest für Zufallsvariable, die nur endlich viele verschiedene Werte annehmen können), er findet sich in jedem Lehrbuch für elementare Stochastik.

Da in unserem Fall Z_i jeweils mit Wahrscheinlichkeit p den Wert 1 annimmt, gilt

$$\mathbf{E}[Z_i] = 0 \cdot \mathbf{P}(Z_i = 0) + 1 \cdot \mathbf{P}(Z_i = 1) = p$$

und deswegen aufgrund der Linearität

$$\mathbf{E}[X] = \mathbf{E}[Z_1] + \cdots + \mathbf{E}[Z_n] = p + \cdots + p = np\ .$$

Das ist bereits die erste Behauptung.

Für die Varianz läuft die Berechnung analog, nur etwas umfänglicher. Es gilt

$$\begin{aligned} \mathbf{E}[(Z_i - p)^2] &= (1-p)^2\mathbf{P}(Z_i = 1) + (0-p)^2\mathbf{P}(Z_i = 0) \\ &= q^2p + p^2q = pq(q+p) = pq \end{aligned}$$

und für $i \neq j$

$$\begin{aligned} &\mathbf{E}[(Z_i - p)(Z_j - p)] \\ &\quad = (1-p)^2\mathbf{P}(Z_i = 1, Z_j = 1) + (0-p)(1-p)\mathbf{P}(Z_i = 0, Z_j = 1) \\ &\qquad + (1-p)(0-p)\mathbf{P}(Z_i = 1, Z_j = 0) + (0-p)^2\mathbf{P}(Z_i = 0, Z_j = 0) \\ &\quad = q^2p^2 - pqqp - qppq + p^2q^2 = 0\ . \end{aligned}$$

Damit folgt, wieder unter Beachtung der Linearität

$$\begin{aligned}
\mathbf{E}\big[(X-np)^2\big] &= \mathbf{E}\big[\big((Z_1-p)+\cdots+(Z_n-p)\big)^2\big] \\
&= \mathbf{E}\Big[\sum_{i=1}^{n}\sum_{j=1}^{n}(Z_i-p)(Z_j-p)\Big] \\
&= \sum_{i=1}^{n}\sum_{j=1}^{n}\mathbf{E}\big[(Z_i-p)(Z_j-p)\big] \\
&= \sum_{i=1}^{n}\mathbf{E}\big[(Z_i-p)^2\big]+\sum_{i=1}^{n}\sum_{j\neq i}\mathbf{E}\big[(Z_i-p)(Z_j-p)\big] \\
&= npq\ .
\end{aligned}$$

Dies ist die Behauptung.

4.3 Kann das Zufall sein? Statistische Tests

Eine Grundidee der Statistik ist es, Daten als Realisierungen von Zufallsvariablen aufzufassen, über deren Wahrscheinlichkeitsverteilung man aus den Daten lernen will. Beim *statistischen Testen* trifft man eine *Hypothese* über die Verteilung und fragt: Liegen die beobachten Daten „im Rahmen“, oder ist hier ein Ereignis eingetreten, das unter der Hypothese so unwahrscheinlich ist, dass wir begründeten Zweifel am Zutreffen der Hypothese hegen sollten? Wir erläutern das Vorgehen an einem Beispiel.

Eine Botschaft ein und desselben Inhalts, es ging um den Vergleich des Erfolgs zweier Therapiemethoden T1 und T2, wurde in zwei unterschiedliche Darstellungsformen verpackt. In Form A wurde herausgestellt, wie groß jeweils der Prozentsatz der Patienten ist, bei denen Behandlung T1 erfolglos bzw. Behandlung T2 erfolgreich war, in Form B wurde der Akzent gerade umgekehrt gesetzt.

Von insgesamt 167 Ärzten, die an einer Sommerschule teilnahmen, wurden rein zufällig 80 ausgewählt, denen die Botschaft in der Form A vermittelt wurde, die restlichen 87 bekamen die Botschaft in der Form B mitgeteilt. Jeder der Ärzte hatte sich daraufhin für die Bevorzugung einer der beiden Therapiemethoden zu entscheiden. Das Ergebnis war:

	für Methode T1	für Methode T2	Summe
A	40	40	80
B	73	14	87
Summe	113	54	167

Die Daten zeigen: In der A-Gruppe gibt es im Verhältnis weniger Befürworter der Therapiemethode T1 als in der B-Gruppe (nämlich 40 : 40 gegen 73 : 14). Haben sich die Ärzte in ihrer Entscheidung durch die Form der Darstellung beeinflussen

lassen? Ein Skeptiker könnte einwenden: „Ach was, auch ohne Beeinflussung kann ein derartiges Ergebnis zustande kommen, wenn der Zufall es will."

Um damit umzugehen, treffen wir folgende Hypothese: Die Form der Botschaft habe keinen Einfluss auf die Meinungsbildung der 167 Ärzte; es wäre so, als ob die einen 80 die Botschaft auf weißem, die anderen 87 eine wörtlich gleichlautende Botschaft auf blauem Papier bekommen hätten. Die Aufteilung der $80+87$ Formulare auf die 113 Befürworter von T1 und die 54 Befürworter von T2 wäre rein zufällig zustande gekommen. Wie wahrscheinlich ist dann eine so extreme Aufteilung wie die beobachtete?

Eine Veranschaulichung: Wenn aus einer Urne mit 80 weißen und 87 blauen Kugeln rein zufällig 113 Kugeln gezogen werden, wie wahrscheinlich ist dann ein so extremes Ergebnis wie das, nur 40 weiße Kugeln zu ziehen?

Die Anzahl X der weißen Kugeln ist eine Zufallsvariable, für die man die Wahrscheinlichkeiten $\mathbf{P}(X = k)$ ohne weiteres berechnen kann. X neißt hypergeometrisch verteilt. Mit $g = 80 + 87 = 167$, $w = 80$, $n = 113$ ergibt sich für ihre Erwartungswert

$$\mathbf{E}[X] = n \cdot \frac{w}{g} = 54.1 \ .$$

Die Wahrscheinlichkeit, ein Ergebnis zu erhalten, das mindestens so weit von 54 weg ist wie der beobachtete Wert 40, ist

$$\mathbf{P}\big(|X - 54| \geq |40 - 54|\big) = \mathbf{P}(X \leq 40) + \mathbf{P}(X \geq 68) = 5.57 \cdot 10^{-6} \ .$$

Wir halten fest: Angenommen die Hypothese trifft zu. Dann tritt ein Ergebnis, das so extrem ist wie das beobachtete, 6 mal in einer Million auf. Damit wird die Hypothese mehr als fragwürdig.

Man nennt die berechnete Wahrscheinlichkeit *den zu den Daten gehörigen p-Wert* oder auch das *beobachtete Signifikanzniveau*, zu dem die Hypothese abgelehnt wird.

Die eben beschriebene Vorgangsweise, bekannt als *Fishers exakter Test auf Unabhängigkeit*, ist ein typisches Beispiel eines *Permutationstests*: Man schreibt die beobachteten Daten in einem Gedankenexperiment dem reinen Zufall zu, indem man jede andere Aufteilung der A-und B-Formulare auf die T1- und T2-Befürworter (jede andere Permutation der 167 Formulare) als ebenso wahrscheinlich ansieht wie die beobachtete.