Wolfgang K. Seiler

Mathematik
und
Information

Vorlesung an der Universität Mannheim
im Herbstsemester 2018

Dieses Skriptum entstand parallel zur Vorlesung und sollte mit möglichst geringer Verzögerung erscheinen. Es ist daher in seiner Qualität auf keinen Fall mit einem Lehrbuch zu vergleichen; insbesondere sind Fehler bei dieser Entstehensweise nicht nur möglich, sondern **sicher**. Dabei handelt es sich wohl leider nicht immer nur um harmlose Tippfehler, sondern auch um Fehler bei den mathematischen Aussagen. Da mehrere Teile aus anderen Skripten für Hörerkreise der verschiedensten Niveaus übernommen sind, ist die Präsentation auch teilweise ziemlich inhomogen.

Das Skriptum sollte daher mit Sorgfalt und einem gewissen Mißtrauen gegen seinen Inhalt gelesen werden. Falls Sie Fehler finden, teilen Sie mir dies bitte persönlich oder per e-mail (seiler@math.uni-mannheim.de) mit. Auch wenn Sie Teile des Skriptums unverständlich finden, bin ich für entsprechende Hinweise dankbar.

Falls genügend viele Hinweise eingehen, werde ich von Zeit zu Zeit Listen mit Berichtigungen und Verbesserungen zusammenstellen. In der online Version werden natürlich alle bekannten Fehler korrigiert.

Biographische Angaben von Mathematikern beruhen größtenteils auf den entsprechenden Artikeln im *MacTutor History of Mathematics archive* (`www-history.mcs.st-andrews.ac.uk/history/`), von wo auch die meisten abgedruckten Bilder stammen. Bei noch lebenden Mathematikern bezog ich mich, soweit möglich, auf deren eigenen Internetauftritt.

Kapitel 0
Was ist Information?

Wir leben bekanntlich im Informationszeitalter, der „Rohstoff Information“ ist ein wesentlicher Wirtschaftsfaktor, und auch für unser Zusammenleben ist Information so wichtig, daß seit einigen Jahrzehnten viele im Gefolge des amerikanischen Mathematikers NORBERT WIENER (1894–1964) und des amerikanischen Soziologen D. BELL (1919–2011) von einer Informationsgesellschaft reden. Was aber ist Information? Und wie kann man sie messen?

Erstaunlicherweise gibt es auf keine dieser beiden Fragen eine allgemein akzeptierte Antwort.

§ 1: Ein Beispiel

Es ist Wahlkampfzeit, und viele Politiker bemühen sich, unsere Stimmen zu bekommen. Deshalb lädt auch Amadeus Wohlgeraten von der Partei für Gesundheit und Wohlstand (PGW) zu einer Informationsveranstaltung, wo er sich und seine Ziele vorstellen möchte. Welche Information erhalten die Teilnehmer dieser Veranstaltung?

Aus der Sicht von Amadeus Wohlgeraten sollen sie lernen, daß nur er und seine Partei sich wirklich für ihre Interessen einsetzen und daß nur sie im Falle eines Wahlsiegs Gesundheit und Wohlstand für alle Bürger bringen werden; die Gegenparteien haben nur Krankheit und Armut zu bieten. Das betrachtet er als die wesentliche Information in seiner Rede.

Seine Anhänger, die alles das schon längst wissen, warten auf die griffigen Slogans, die die PGW für solche Zwecke entwerfen ließ, um an den

richtigen Stellen ihre Begeisterung zu zeigen; als einzige Information nehmen sie mit nach Hause, daß sie die Stimmung im Saal dominierten.

Der Redakteur der Lokalzeitung mußte im Laufe seines Lebens schon viel zu viele Wahlversammlungen besuchen; er kann sich bereits im Voraus ziemlich genau denken, worum es in der Rede gehen wird. Ihn interessiert nur, ob Amadeus Wohlgeraten auf dem Weg zum Podium stolpert, ob es lustige Versprecher gibt oder, idealerweise, eine Saalschlacht; diese Information genügen, um seinem bereits eine Woche zuvor geschriebenen Bericht die endgültige Form zu geben.

Das Modehaus, das zu den Sponsoren der Partei für Gesundheit und Wohlbefinden zählt, ist vertreten durch den für die Kundenzeitschrift zuständigen Mitarbeiter. Was er über die gesunden und günstigen Stoffqualitäten der angebotenen Waren schreiben wird, ist natürlich unabhängig vom Verlauf der Versammlung; er möchte aber zumindest noch erwähnen, welchen Anzug mit optimal passender Krawatte Amadeus Wohlgeraten für diesen Abend aus dem großen Angebot des Modehauses ausgewählt hat.

Der Vorsitzende des örtlichen Vereins der Kaulquappenfreunde kann sich nicht erklären, wie sich jemand mit Politik beschäftigen kann, obwohl es noch so viele offene Fragen über Kaulquappen gibt. Trotzdem muß er alle Wahlveranstaltungen besuchen, denn wer auch immer die Wahl gewinnt, wird seine Kaulquappenpolitik möglicherweise danach ausrichten, ob er sich von den Kaulquappenfreunden unterstützt fühlt oder nicht. Den Vortrag faßt er kurz als „übliches Politikergeschwätz“ zusammen; doch in der nächsten Ausgabe des *Kaulquappenfreunds* kann er in den *Informationen aus dem Vorstand* stolz vermelden, daß er mit dem PGW-Kandidaten Amadeus Wohlgeraten über die wichtige Rolle der Kaulquappenzucht für Gesundheit und Wohlstand der Bevölkerung gesprochen habe.

Die Partei für Sorgenfreies Wohlbefinden, einer der Hauptkonkurrenten der Partei für Gesundheit und Wohlstand, möchte Wohlgeratens Rede natürlich genau analysieren; da jegliche Art von Ton- und Videoaufnahmen verboten sind, schicken sie einen Stenographen, der die Rede Wort für Wort mitschreibt. Da dieser zum Mitdenken keine Zeit hat, ist für

ihn der Informationsgehalt der Veranstaltung einfach die Folge der zu notierenden Worte.

Der Organisator einer Wahlwette möchte wissen, wie sich die Wahlchancen von Amadeus Wohlgeraten durch die Veranstaltung verändern, so daß er die Wettquoten gegebenenfalls neu festlegen kann. Die Information, die er mitnimmt, ist eine neue Schätzung für die Wahrscheinlichkeit eines Siegs der Partei für Gesundheit und Wohlbefinden, basiert auf die bekannten Umfrageergebnisse und seine Einschätzung von Stimmungswandeln im Saal.

Der Mathematiker, der sich mit Information beschäftigt, darf sich nicht darauf beschränken, dieses Geschehen einfach in einen mehr oder weniger nützlichen Formalismus zwingen; er möchte *quantitativ* beschreiben, was hier geschehen ist; zumindest für den Wettanbieter sollte es sogar möglich sein, die gewonnene Information direkt in Euro und Cent umzurechnen.

Angesichts der Vielzahl von Interessen der Beteiligten wird es dabei sicherlich nicht reichen, die an diesem Abend vermittelte Information durch eine einzige Zahl zu beschreiben; bei jedem einzelnen müssen wir sowohl sein Vorwissen als auch seinen Umgang mit dem Gesagten berücksichtigen. Was bei ihm ankommt, wurde möglicherweise bereits einigen Verarbeitungsschritten entworfen, z.B. weil er dank des Lärmpegels nur einen Teil der Rede hören kann, und auch er verarbeitet die ankommende Information weiter (War von Kaulquappen die Rede?), bevor ein Teil des Ergebnisses in sein Gedächtnis wandert. Schon jetzt können wir einen wesentlichen Aspekt dieser Informationsverarbeitung festhalten: Wie auch immer wir Information quantitativ fassen werden muß offensichtlich gelten, daß sie durch diese Verarbeitung höchstens abnehmen kann. Das heißt allerdings nicht unbedingt, daß sie dadurch weniger nützlich werden *muß:* Eine ungeordnete Sammlung von mehreren Millionen Datensätzen ist oft deutlich weniger nützlich als ein Satz von daraus abgeleiteten statistischen Kenngrößen. Die Information darüber ist zwar natürlich in den Datensätzen enthalten, sie zu extrahieren kann aber aufwendig sein. Auch mit solchen Fragen muß sich ein Mathematiker beschäftigen.

Am einfachsten zu fassen ist wohl noch die Information aus Sicht des Stenographen: In erster Näherung könnten wir einfach zählen, wie viele Zeichen er zu Papier gebracht hat. Aber selbst das ist nicht wirklich wohldefiniert, denn Stenographen arbeiten schließlich auch mit Kürzeln, die verwendet werden können, aber nicht müssen, und wenn sich Amadeus Wohlgeraten zu oft wiederholt haben sollte, erfand der Stenograph vielleicht auch noch ad hoc neue Kürzel für einige besonders häufige Phrasen. Die Frage, wie weit der dabei optimieren kann, führt uns zur SHANNONschen Informationstheorie und, in letzter Konsequenz, zur algorithmischen Informationstheorie.

Um das Vorwissen des Lokalredakteurs ins Spiel zu bringen, können wir die Information, die er bereits vor der Veranstaltung hatte, vergleichen mit seinem Informationsstand danach; die Differenz ist die neu gewonnene Information. Dies führt uns auf den Begriff der bedingten Information.

Im Falle des Buchmachers müssen wir zwei Wahrscheinlichkeitsverteilungen miteinander vergleichen: Die Siegwahrscheinlichkeiten der einzelnen Kandidaten so, wie er sie vor der Veranstaltung einschätzte, und die entsprechenden Zahlen, nach denen er künftig seine Prämien berechnet. Die gewonnene Information aus seiner Sicht ist somit eine Art Distanz zwischen zwei Wahrscheinlichkeitsverteilungen, die wir später als KULLBACK-LEIBLER-Distanz formalisieren werden; der finanzielle Wert der Information läßt sich über die Erwartungswerte für seinen Gewinn bezüglich der beiden Verteilungen quantifizieren.

Die Analysten in der Zentrale der Partei für Sorgenfreies Wohlbefinden werten nicht nur den (nach der Veranstaltung in ihren Computer übertragenen) Bericht des Stenographen aus, sondern zahlreiche weitere Berichte von ähnlichen Veranstaltungen. Sie müssen einerseits diese Berichte nach Gemeinsamkeiten gruppieren, um so einen Überblick über die gegnerische Strategie zu bekommen; andererseits müssen sie aber auch Ausreißer finden, die sich vielleicht als Wahlkampfmunition eignen könnten. Da sie auch die entsprechenden Daten anderer politischer Gegner auswerten müssen, haben sie viel zu tun und wollen ihre Arbeit möglichst automatisieren. Die mathematischen Verfahren, die

sie dabei anwenden können, werden uns im zweiten Teil der Vorlesung beschäftigen.

§2: Information aus sprachlicher Sicht

Die Etymologie des Wortes *Information* trägt leider nur wenig zum Verständnis dieses Begriffs bei: Das lateinische *informatio* enthält den Wortstamm *forma,* Form oder Gestalt, und informatio wurde im Sinne von Bildung oder Unterricht gebraucht. Bei der Aufnahme des Worts in die deutsche Sprache im 15. bis 16. Jahrhundert verschob sich die Bedeutung zu *Nachricht* oder *Unterrichtung* (über einen Sachverhalt), und diesen Sinn hat das Wort heute auch in anderen modernen Sprachen.

Information hat also etwas mit der Übermittlung von Nachrichten zu tun; eine mathematische Theorie der Information muß sich daher insbesondere auch mit der Struktur von Nachrichten beschäftigen. Dafür interessiert sich selbstverständlich nicht nur die Mathematik; schon in der Logik von ARISTOTELES (384–322) finden sich Überlegungen, die in diese Richtung gehen, und auch die Grammatiker befassen sich schon seit weit über Tausend Jahren mit entsprechenden Fragen.

Einen wesentlichen Schritt in Richtung auf eine mathematische Beschreibung von Sprache leistete 1879 der Philosoph FRIEDRICH LUDWIG GOTTLOB FREGE (1848–1925) mit seiner *Begriffsschrift,* in der er die mathematische Logik in ihrer heutigen Form begründete. Sein Versuch, die gesamte Mathematik auf Logik zu reduzieren, scheiterte zwar, führte aber zur Entwicklung alternativer Ansätze sowohl zur Grundlegung der Mathematik als auch zur allgemeinen Untersuchung formaler Systeme und schließlich auch natürlicher Sprachen.

Vor allem durch die Arbeiten des amerikanischen Philosophen CHARLES WILLIAM MORRIS (1901–1979) entstand ab Ende der Dreißigerjahre die *Semiotik* als allgemeine Lehre von den Zeichen und ihrer Verwendung. Er unterscheidet drei Aspekte:

1. Die Syntax, in der es um die Zeichen selbst und die Regeln für ihre Aneinanderreihung geht.

2. Die Semantik, die sich mit der Bedeutung von Zeichenfolgen beschäftigt.

3. Die Pragmatik, in der es um deren *Gebrauch* geht: Dazu zählen beispielsweise unterschiedliche Bedeutungsebenen (Kopf, Haupt, Rübe), aber auch unterschiedliche Ziele, die mit einem Wort oder Satz erreicht werden sollen: Der Ausruf *Feuer!* etwa kann zwar bedeuten, daß es brennt, kann aber beim Militär auch der Befehl zum Schießen sein und bei einem Raucher die Bitte, ihm die Zigarette anzuzünden.

Die klassische Informationstheorie beschränkt sich auf rein syntaktische Aspekte; bei der Suche nach Information stehen dagegen semantische Aspekte im Vordergrund. Pragmatik spielt bei der mathematischen Behandlung von Information bislang keine Rolle.

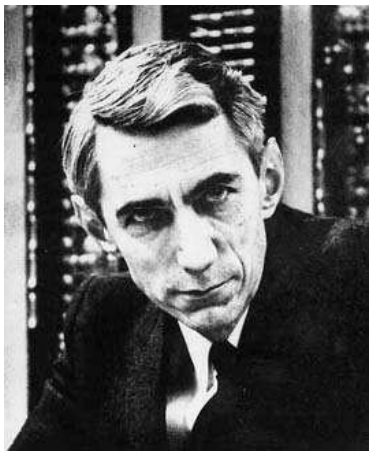
Sobald wir von praktischen Anwendungen der Information reden, kommt allerdings ein neuer, bislang noch nicht erwähnter Gesichtspunkt ins Spiel: Information kann einen teilweise sogar beträchtlichen wirtschaftlichen Wert darstellen. Informationen über den Zustand eines Landes oder einen Unternehmens beeinflussen beispielsweise die Preise von Aktien, und je nachdem wie früh oder spät jemand darauf reagiert, kann er viel Geld gewinnen oder verlieren.

Der Wert solcher Informationen kann allerdings nicht objektiv beziffert werden: Die Nachricht, daß in einer südafrikanischen Goldmine eine neue stark erzführende Ader entdeckt wurde, ist wertlos für jemanden, der kein Geld für Aktienkäufe hat oder grundsätzlich nur in Europa investiert; für einen südafrikanischen Investor dagegen kann die Information einen beträchtlichen Wert haben.

Auch für ihn kann eine entsprechende Meldung allerdings völlig wertlos sein, etwa weil er sie schon seit Tagen kennt und längst darauf reagiert hat. Wenn wir vom Wert einer Information sprechen wollen, kann es sich daher immer nur um den Wert für eine bestimmte Person mit bestimmten Interessen handeln; insbesondere ist deren Vorwissen ein wichtiger, wenn auch bei weitem nicht der einzige Aspekt. Wir werden daher eine ganze Reihe weiterer Maße benötigen, um auch solche Situationen mathematisch zu beschreiben.

Kapitel 1
Shannons Informationstheorie

Die bekannteste quantitative Definition von Information geht zurück auf CLAUDE SHANNON; Bücher mit Titeln wie *Informationstheorie* befassen sich meist ausschließlich damit. Die inhaltliche Interpretation von Information spielt hier keinerlei Rolle, es geht nur um ihre sichere Übermittlung. Sicherheit bezieht sich dabei sowohl auf den Schutz vor Übertragungsfehlern (durch fehlererkennende und -korrigierende Codes) als auch auf die Geheimhaltung (Kryptographie).



CLAUDE ELWOOD SHANNON (1916–2001) wurde in Petoskey im US-Bundesstaat Michigan geboren; 1936 verließ er die University of Michigan mit sowohl einem Bachelor der Mathematik als auch einem Bachelor der Elektrotechnik, um am M.I.T. weiterzustudieren. Seine 1938 geschriebene Diplomarbeit *A symbolic analysis of relay and switching circuits* bildet die Grundlage der digitalen Informationsverarbeitung auf der Grundlage der hier entwickelten Schaltlogik; seine Dissertation 1940 befaßte sich mit Anwendungen der Algebra auf die MENDELschen Gesetze. Danach arbeitete er bis 1956 bei den Bell Labs, wo er während des zweiten Weltkriegs insbesondere über die Sicherheit kryptographischer Systeme forschte. Seine *Mathematical theory of cryptography* wurde aus Geheimhaltungsgründen erst 1949 zur Veröffentlichung freigegeben. Seine wohl bekannteste Arbeit ist die 1948 erschienene *Mathematical theory of communication,* in der er die fehlerfreie Übertragung von Nachrichten über einen gestörten Kanal untersuchte. Von 1956 bis zu seiner Emeritierung 1978 lehrte er am M.I.T., das er dadurch zur führenden Universität auf dem Gebiet der Informationstheorie und Kommunikationstechnik machte. Zu seinen zahlreichen Arbeiten zählt auch eine über die mathematische Theorie der Jongliermuster, anhand derer Jongleure eine Reihe neuer Muster gefunden haben; auch konstruierte er mehrere Jonglierroboter.

§1: Die Entropie einer Quelle

Gerade weil der SHANNONsche Informationsbegriff der in den Wissenschaften am weitesten verbreitete ist, müssen wir uns als allererstes klar werden, was er nicht ist: Es geht nicht darum, den Informationsgehalt einer einzelnen Nachricht zu messen.

Im 1949 erschienenen Buch *The mathematical theory of communication,* das SHANNONs gleichnamige Arbeit von 1948 zusammen mit einer ausführlichen Einleitung von WARREN WEAVER enthält, schreibt letzterer zu Beginn von §2.2:

> The word *information,* in this theory, is used in a special sense that must not be confused with its ordinary usage. In particular, *information* must not be confused with meaning.
>
> In fact, two messages, one of which is heavily loaded with meaning and the other of which is pure nonsense, can be exactly equivalent, from the present viewpoint, as regards information. It is this, undoubtedly, that Shannon means when he says that "'the semantic aspects are necessarily irrelevant to the engineering aspects."' But this does not mean that the engineering aspects are necessarily irrelevant to the semantic aspects.
>
> To be sure, this word information in communication theory relates not so much to what you *do* say, as in what you *could* say.

SHANNON betrachtet Nachrichten also stets vor dem Hintergrund einer Auswahl; was er messen will, sind die Wahlmöglichkeiten des Senders und die Ungewißheit des Empfängers *vor* Übermittlung der Nachricht.

Ein solcher Ansatz kann nur funktionieren, wenn sowohl für den Sender als auch den Empfänger klar ist, welche Nachrichten grundsätzlich übertragen werden *könnten.* Zu diesem Zweck geht SHANNON aus von einem festen *Alphabet* A. Darunter versteht er irgendeine endliche Menge, deren Elemente zwar als Buchstaben bezeichnet werden, die aber auch elektrische Signale, ASCII-Zeichen, Ereignisse und vieles andere sein können.

Der Sender wird modelliert durch eine Nachrichten*quelle,* die eine Folge von Buchstaben des Alphabets A produziert. In den seltensten Fällen werden dabei alle Buchstaben mit der gleichen Häufigkeit vorkommen; in SHANNONs Ansatz ist daher jedem Buchstaben $x_i \in A$ eine

Häufigkeit p_i zugeordnet. Natürlich müssen alle $p_i \geq 0$ sein und ihre Summe gleich eins; bei einem Alphabet aus n Buchstaben liegt das Tupel der Wahrscheinlichkeiten also in der Menge

$$\Delta_n = \left\{ (p_1, \ldots, p_n) \;\middle|\; p_i \geq 0 \text{ für alle } i \text{ und } \sum_{i=1}^{n} p_i = 1 \right\}.$$

Mathematisch gesehen ist eine Nachrichtenquelle also eine diskrete Zufallsvariable, die Werte aus A annimmt.

Ausgangspunkt für die Quantifizierung von Information ist der *mittleren* Informationsgehalt eines Buchstabens. Da dieser Informationsgehalt sicherlich nicht von den Namen der Buchstaben abhängt, können wir diesen einfach als eine Funktion der Buchstabenwahrscheinlichkeiten p_i betrachten; wir suchen also für jede natürliche Zahl n eine Funktion $H_n \colon \Delta_n \to \mathbb{R}_{\geq 0}$, so daß $H_n(p_1, \ldots . p_n)$ der mittlere Informationsgehalt eines Buchstabens aus einem n-elementigen Alphabet ist. wobei $p_1, \ldots, p_n$ die Häufigkeiten der einzelnen Buchstaben sind.

Eine solche Funktion sollte nach SHANNON vernünftigerweise die folgenden Bedingungen erfüllen:

1. Alle H_n sind stetig, denn natürlich sollen kleine Änderungen an den p_i nicht zu sprunghaften Änderungen am Informationsgehalt führen.
2. Die Funktion $L(n) = H_n\left(\frac{1}{n}, \ldots, \frac{1}{n}\right)$ ist monoton wachsend, d.h. wenn wir eine Quelle haben, die die Buchstaben ihres Alphabets mit gleicher Wahrscheinlichkeit ausstößt, steigt der Informationsgehalt pro Buchstabe mit der Buchstabenanzahl. Beispielsweise liefert ein Meßfühler mehr Information, wenn er eine größere Auflösung hat.

Etwas technischer und schwerer zu verstehen ist SHANNONs dritte Forderung: Vor jeder Übertragung eines Buchstabens steht der Sender vor einer Wahl. Wenn er diese Wahl in mehrere Teilentscheidungen zerlegt, soll sich dadurch nichts an der Gesamtinformation ändern. Konkret: Ist C eine Teilmenge des Alphabets A, so kann der Sender in einem ersten Schritt entweder ein Element von $A \setminus C$ auswählen oder sich dafür entscheiden, ein Element aus C zu senden. Im letzteren Fall muß er dann in einem zweiten Schritt konkretisieren, welches der Elemente aus C er senden will. Wenn wir der Einfachheit halber annehmen, daß $A \setminus C$

die ersten m der n Elemente von A enthält und die Summe der Wahrscheinlichkeiten für die Elemente aus C gleich p^* ist, soll dann also gelten

3. $H_n(p_1,\ldots,p_n) = H_{m+1}(p_1,\ldots,p_m,p^*)+p^* H_{n-m}\left(\frac{p_{m+1}}{p^*},\ldots,\frac{p_n}{p^*}\right)$, falls $p^* = \sum_{i=m+1}^{n} p_i > 0$ ist. (Der Faktor p^* vor dem zweiten Summanden kommt daher, daß nur mit Wahrscheinlichkeit p^* überhaupt eine zweite Entscheidung getroffen wird, und die Nenner in den Argumenten sind notwendig, da der Buchstabe a_i mit $i > m$ die Wahrscheinlichkeit p_i/p^* hat, falls bereits feststeht, daß ein Buchstabe aus C gesendet wird.)

Vielleicht hilft der folgende Spezialfall, diese Bedingung etwas besser zu verstehen:

Lemma: $A = \{a_1,\ldots,a_m\}$ und $B = \{b_1,\ldots,b_n\}$ seien zwei endliche Alphabete, wobei a_i mit Wahrscheinlichkeit p_i und b_j mit Wahrscheinlichkeit q_j auftrete. Gibt man dem Element $(a_i,b_j) \in A \times B$ die Wahrscheinlichkeit $p_i q_j$, so ist

$$H_{nm}(\ldots,p_i q_j,\ldots) = H_m(p_1,\ldots,p_m) + H_n(q_1,\ldots,q_n)\,,$$

bei zwei unabhängigen Quellen addieren sich also die mittleren Informationsgehalte.

Beweis: Wir wenden Forderung 3 an auf die Teilmenge $C = \{a_m\} \times B$ von $A \times B$. Hier ist $p^* = \sum_{j=1}^{n} p_m q_j = p_m$, also folgt

$$H_{nm}(\underbrace{\ldots,p_i q_j,\ldots}_{\substack{i=1,\ldots,m\\ j=1,\ldots,n}}) = H_{(m-1)n+1}(\underbrace{\ldots,p_i q_j,\ldots}_{\substack{i=1,\ldots,m-1\\ j=1,\ldots,n}},p_m)+p_m H_n(q_1,\ldots,q_n)\,.$$

Auf den ersten Summanden links können wir das gleiche Argument anwenden und die Paare mit a_{m-1} abspalten *usw.;* wir erhalten schließlich

$$\begin{aligned} H(\underbrace{\ldots,p_i q_j,\ldots}_{\substack{i=1,\ldots,m\\ j=1,\ldots,n}}) &= H(p_1,\ldots,p_m) + \sum_{i=1}^{m} p_i H(q_1,\ldots,q_n) \\ &= H(p_1,\ldots,p_m) + H(q_1,\ldots,q_n)\,. \end{aligned}$$

■

Satz: Zu jeder Familie von Funktionen $H\colon \Delta_n \to \mathbb{R}_{\geq 0}$, die obige drei Bedingungen erfüllt, gibt es eine reelle Zahl $a > 1$ so daß

$$H(p_1, \ldots, p_n) = -\sum_{i=1}^{n} p_i \log_a p_i$$

ist, wobei $p_i \log p_i$ für $p_i = 0$ als Null interpretiert werden soll. Insbesondere ist H bis auf einen positiven Faktor eindeutig bestimmt.

Beweis: In einem *ersten Schritt* beschränken wir uns auf den Fall, daß alle p_i gleich sind, betrachten also für jedes $n \in \mathbb{N}$ nur den einen Wert $L(n) = H\left(\frac{1}{n}, \ldots, \frac{1}{n}\right)$.

A_1 bis A_m seien m voneinander unabhängige Quellen, die jeweils r verschiedene Buchstaben mit gleicher Wahrscheinlichkeit $1/r$ liefern; der Informationsgehalt jeder dieser Quellen ist also $L(r)$. Das Produkt $A_1 \times \cdots \times A_m$ enthält r^m tupel, die allesamt mit derselben Wahrscheinlichkeit $1/r^m$ auftreten; die Gesamtinformation ist also

$$H\left(\frac{1}{r^m}, \ldots, \frac{1}{r^m}\right) = L(r^m)\,.$$

Aus dem gerade bewiesenen Lemma folgt induktiv, daß dies die Summe der Informationsgehalte der Quellen A_i ist, d.h. $L(r^m) = mL(r)$.

Nun betrachten wir natürliche Zahlen r, s, m, n mit $r^m \leq s^n \leq r^{m+1}$; dann ist (unabhängig von der Basis des Logarithmus)

$$m \log r \leq n \log s \leq (m+1) \log r \quad \text{oder} \quad \frac{m}{n} \leq \frac{\log s}{\log r} \leq \frac{m+1}{n}\,.$$

Wegen der in Forderung zwei postulierten Monotonie von L gilt die Ungleichung $L(r^m) \leq L(s^n) \leq L(r^{m+1})$; wie wir gerade gesehen haben, können wir diese auch schreiben als

$$mL(r) \leq nL(s) \leq (m+1)L(r)\,.$$

Division durch $nL(r)$ macht daraus

$$\frac{m}{n} \leq \frac{L(s)}{L(r)} \leq \frac{m+1}{n}\,,$$

$L(s)/L(r)$ und $\log s/\log r$ liegen daher beide im Intervall $\left[\frac{m}{n}, \frac{m+1}{n}\right]$, so daß

$$\left|\frac{L(s)}{L(r)} - \frac{\log s}{\log r}\right| \leq \frac{1}{n}$$

sein muß. Da n beliebig groß gewählt werden kann, gilt dies für alle $n \in \mathbb{N}$, d.h.

$$\frac{L(s)}{L(r)} = \frac{\log s}{\log r} \quad \text{oder} \quad L(s) = \frac{L(r)}{\log r} \cdot \log s\,.$$

Somit ist $L(s)$ proportional zu einem Logarithmus, wobei die Proportionalitätskonstante $L(r)/\log r$ wegen der Monotonie sowohl von L als auch des Logarithmus positiv sein muß. Mithin gibt es eine reelle Zahl $a > 1$ mit $L(n) = \log_a n$ für alle $n \in \mathbb{N}$.

Im speziellen Fall von Quellen, die alle Buchstaben mit gleicher Wahrscheinlichkeit ausgeben, ist der Satz damit bewiesen.

Im *zweiten Schritt* verlangen wir von den Wahrscheinlichkeiten p_i nur noch, daß es sich dabei um positive rationale Zahlen handelt. Wir betrachten also ein Alphabet $A = \{a_1, \ldots, a_n\}$ aus n Buchstaben, deren i-ter die Wahrscheinlichkeit $p_i = g_i/g$ habe mit $g_i \in \mathbb{N}$. Da die Summe aller p_i gleich eins ist, muß dabei $\sum g_i = g$ sein.

Weiter betrachten wir ein Alphabet B aus g Buchstaben $b_1, \ldots, b_g$, die allesamt die gleiche Wahrscheinlichkeit $1/g$ haben. Diese Buchstaben verteilen wir auf n disjunkte Teilmengen $B_i \subseteq B$ derart, daß B_i aus g_i Buchstaben besteht. Durch n-fache Anwendung der dritten Forderung erhalten wir die Gleichung

$$H\left(\frac{1}{g}, \ldots, \frac{1}{g}\right) = H(p_1, \ldots, p_n) + \sum_{i=1}^{n} p_i H\left(\frac{1}{g_i}, \ldots, \frac{1}{g_i}\right).$$

Die Funktionen mit lauter gleichen Argumenten können wir durch Lo-

garithmen ausdrücken und erhalten dann

$$\begin{aligned} H(p_1,\dots,p_n) &= H\Big(\frac{1}{g},\dots,\frac{1}{g}\Big) - \sum_{i=1}^{n} p_i H\Big(\frac{1}{g_i},\dots,\frac{1}{g_i}\Big) \\ &= \log_a g - \sum_{i=1}^{n} p_i \log_a g_i = \sum_{i=1}^{n} p_i \big(\log_a g - \log_a g_i\big) \\ &= -\sum_{i=1}^{n} p_i \log_a \frac{g_i}{g} = -\sum_{i=1}^{n} p_i \log_a p_i \,, \end{aligned}$$

wie behauptet.

Als *dritten Schritt* betrachten wir den allgemeinen Fall. Wir gehen also aus von einer Familie von Funktionen $H\colon \Delta_n \to \mathbb{R}$, die alle drei Forderungen SHANNONs erfüllt. Wie wir bereits wissen, ist dann

$$H(p_1,\dots,p_n) = -\sum_{i=1}^{n} p_i \log p_i \,,$$

falls wir für alle p_i positive rationale Zahlen einsetzen. Auf der rechten Seite steht eine Funktion, die für alle positiven reellen Werte der p_i stetig ist; die Funktion links muß nach SHANNONs erster Forderung stetig auf Δ_n sein. Da zwei stetige Funktionen, die für alle rationalen Werte aus einer offenen Menge übereinstimmen, dort gleich sind, gilt obige Gleichung im Innern von Δ_n, also für alle positiven reellen Werte der p_i.

Bleibt noch der Fall, daß eines oder mehrere der p_i verschwinden. In diesem Fall ist die rechte Seite nicht definiert, denn die Logarithmusfunktion hat an der Stelle Null einen Pol. Im Satz war vereinbart, daß wir für $p_i = 0$ den Term $p_i \log p_i$ als Null interpretieren; wenn wir zeigen können, daß dadurch die Funktion stetig auf Δ_n fortgesetzt wird, folgt Gleichheit auch in diesem Fall.

Offenbar genügt es, einen einzelnen Summanden zu betrachten; nach der Regel von DE L'HÔPITAL ist für den natürlichen Logarithmus

$$\lim_{p\searrow 0} p\log p = \lim_{p\searrow 0} \frac{\log p}{1/p} = \lim_{p\searrow 0} \frac{1/p}{-1/p^2} = \lim_{p\searrow 0}(-p) = 0 \,,$$

und da jeder andere Logarithmus proportional zum natürlichen ist, haben wir diesen Grenzwert auch für Logarithmen zu einer beliebigen Basis. Damit ist der Satz vollständig bewiesen.

■

Damit sind wir allerdings noch nicht ganz fertig: Zwar wissen wir nun, daß jede Funktion, die SHANNONs drei Bedingungen genügt, die angegebene Form haben muß, wir wissen aber noch nicht, ob es überhaupt solche Funktionen gibt. Dazu müssen wir noch nachprüfen, daß die Funktionen

$$H(p_1, \ldots, p_n) = -\sum_{i=1}^{n} p_i \log_a p_i$$

alle drei Bedingungen erfüllen.

Die Stetigkeit ist klar, da H nur durch Grundrechenarten und Logarithmen definiert ist. Auch mit der zweiten Bedingung gibt es keine Probleme, denn

$$L\left(\frac{1}{n}\right) = H\left(\frac{1}{n}, \ldots, \frac{1}{n}\right) = -\sum_{i=1}^{n} \frac{1}{n} \log \frac{1}{n} = \log n$$

ist eine monoton wachsende Funktion. Die dritte Bedingung schließlich ist erfüllt, denn für $m < n$ und $p^* = p_{m+1} + \cdots + p_n$ ist

$$\begin{aligned} H(p_1, \ldots, p_n) &= -\sum_{i=1}^{n} p_i \log_a p_i \\ &= -\sum_{i=1}^{m} p_i \log_a p_i - p^* \log_a p^* + p^* \log_a p^* - \sum_{i=m+1}^{n} p_i \log_a p_i \\ &= H(p_1, \ldots, p_m, p^*) + \sum_{i=m+1}^{n} p_i (\log_a p^* - \log_a p_i) \\ &= H(p_1, \ldots, p_m, p^*) - \sum_{i=m+1}^{n} p_i \log_a \frac{p_i}{p^*} \\ &= H(p_1, \ldots, p_m, p^*) + p^* H\left(\frac{p_{m+1}}{p^*}, \ldots, \frac{p_n}{p^*}\right). \end{aligned}$$

Damit haben wir für die Definition des Informationsgehalts nur noch die Freiheit, die Basis a des Logarithmus festzulegen; die traditionelle Wahl ist $a = 2$.

Definition: Die Entropie einer Quelle A mit einem m-buchstabigen Alphabet und Wahrscheinlichkeit p_i für das Auftreten des i-ten Buchstaben ist

$$H(A) = -\sum_{i=1}^{m} p_i \log_2 p_i \,.$$

Der Name *Entropie* ist ein Kunstwort, das der deutsche Physiker RUDOLF CLAUSIUS (1822–1888) in seiner Arbeit

> R. CLAUSIUS: Über verschiedene für die Anwendung bequeme Formen der Hauptgleichungen der Wärmetheorie, *Annalen der Physik und Chemie* **125** (1865), 353-400

einführte. Auf Seite 390 schreibt er:

> Sucht man für S einen bezeichnenden Namen, so könnte man . . . von der Größe S sagen, sie sey der *Verwandlungsinhalt* des Körpers. Da ich es aber für besser halte, die Namen derartiger für die Wissenschaft wichtiger Größen aus den alten Sprachen zu entnehmen, damit sie unverändert in allen neuen Sprachen angewandt werden können, so schlage ich vor, die Größe S nach dem griechischen Worte ἡ τροπή, die Verwandlung, die Entropie des Körpers zu nennen. Das Wort *Entropie* habe ich absichtlich dem Wort *Energie* möglichst ähnlich gebildet, denn die beiden Größen, welche durch diese Worte benannt werden sollen, sind ihren physikalischen Bedeutungen nach einander so nahe verwandt, daß eine gewisse Gleichartigkeit in der Benennung mir zweckmäßig zu seyn scheint.

Die Größe S, von der er hier spricht, hilft unter anderem bei der Erklärung, warum Wärme nie von einem kälteren zu einem wärmeren Körper fließen kann; wie LUDWIG BOLTZMANN (1844-1906) später gezeigt hat, kann sie auch mikroskopisch definiert werden durch eine Formel, die eng mit der hier zu definierenden SHANNONschen Entropie verwandt ist.

Als erstes Beispiel betrachten wir eine Zufallsvariable X, die alle Werte aus dem Alphabet A mit gleicher Wahrscheinlichkeit annimmt. Falls A aus n Buchstaben besteht, ist also $p(a) = 1/n$ für alle $a \in A$ und damit

$$H(X) = -\sum_{a\in A} p(a) = -\sum_{a\in A} \frac{1}{n} \log_2 \frac{1}{n} = -\log_2 \frac{1}{n} = \log_2 n \,.$$

Speziell im Fall einer Zweierpotenz $n = 2^r$ ist das gleich r und entspricht der Tatsache, daß man 2^r Objekte durch r Binärziffern eindeutig bezeichnen kann.

Im Falle eines Alphabets $A = \{a, b, c, d, e\}$ aus fünf Buchstaben und einer Zufallsvariablen Y, die diese mit Wahrscheinlichkeiten $p(a) = \frac{1}{2}$

und $p(b) = p(c) = p(d) = p(e) = \frac{1}{8}$ annimmt, ist

$$H(Y) = -\frac{1}{2}\log_2\frac{1}{2} - 4\cdot\frac{1}{8}\log\frac{1}{8} = \frac{1}{2} + \frac{4}{8}\cdot 3 = 2\,,$$

aber natürlich gibt es keine Möglichkeit, fünf Buchstaben mit nur zwei Binärziffern zu bezeichnen. Mit der Kodierung

$$a = 0, \quad b = 100, \quad c = 101, \quad d = 110 \quad \text{und} \quad e = 111$$

kommen wir aber immerhin *im Durchschnitt* mit zwei Binärziffern aus, denn in der Hälfte aller Fälle haben wir a, wofür eine Ziffer ausreicht, und in der anderen Hälfte der Fälle brauchen wir drei Buchstaben, im Mittel also zwei. Für eine Zufallsvariable Z, die jedes Element von A mit Wahrscheinlichkeit $\frac{1}{5}$ annimmt, ist dagegen $H(Z) = \log_2 5 \approx 2{,}321928095$, und hier gibt es offensichtlich *keine* Kodierung, bei der wir im Durchschnitt $H(Z)$ Binärziffern brauchen, denn das arithmetische Mittel aus fünf natürlichen Zahlen muß ein Vielfaches von $\frac{1}{5}$ sein. Die nächstgrößere Zahl mit dieser Eigenschaft wäre 2,4, und das können wir tatsächlich erreichen, zum Beispiel mit der Kodierung

$$a = 00, \quad b = 01, \quad c = 10, \quad d = 110 \quad \text{und} \quad e = 111\,.$$

SHANNONs Entropiebegriff steht also offensichtlich im Zusammenhang mit der mittleren Anzahl von Binärziffern, mit der wir die Buchstaben aus dem Alphabet kodieren können; wie genau dieser Zusammenhang aussieht, werden wir in Kürze untersuchen.

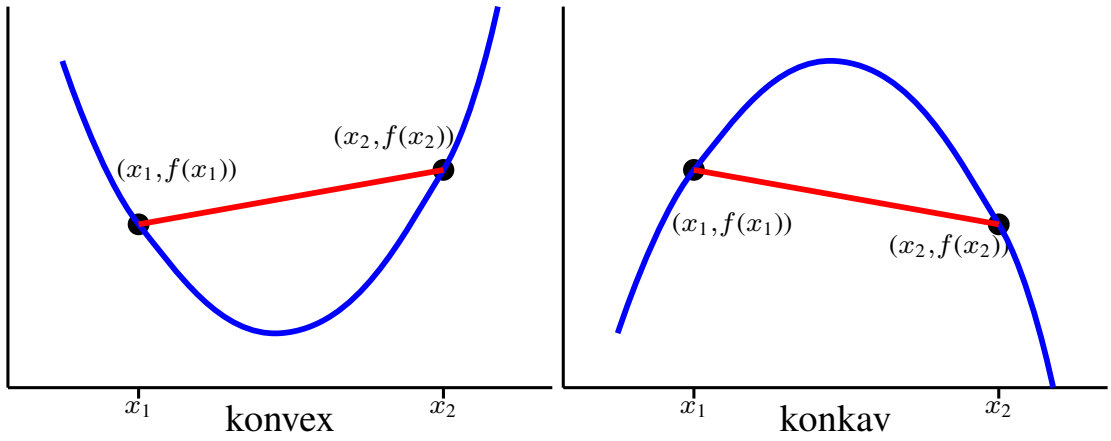
§2: Konvexität

SHANNONs drei Forderungen reichen zwar aus, um die Entropie (bis auf eine positive Konstante) eindeutig zu charakterisieren; der Begriff wäre aber nicht sonderlich nützlich, wenn wir nicht noch eine ganze Reihe weiterer Aussagen herleiten könnten. So erwarten wir beispielsweise, daß eine Quelle, die einen bestimmten ihrer Buchstaben mit einer sehr hohen Wahrscheinlichkeit produziert, einen kleineren mittleren Informationsgehalt hat als eine, bei der alle Buchstaben mit ungefähr gleicher Wahrscheinlichkeit vorkommen. Mit Aussagen dieser Art werden wir es auch noch in anderen Zusammenhängen zu tun haben; deshalb lohnt es sich, das Problem etwas allgemeiner anzugehen.

Definition: *a)* Eine *Teilmenge* $\Delta \subseteq \mathbb{R}^n$ heißt *konvex,* wenn zu je zwei Punkten $P, Q \in \Delta$ und jede reelle Zahl λ aus dem abgeschlossenen Intervall $[0,\ 1]$ auch der Punkt $(1-\lambda)P + \lambda Q$ in Δ liegt, wenn Δ also mit je zwei Punkten auch deren Verbindungsstrecke enthält.
b) Eine *Funktion* $f\colon \Delta \to \mathbb{R}$ auf einer konvexen Teilmenge $\Delta \subseteq \mathbb{R}^n$ heißt *konvex,* wenn für je zwei Punkte $P, Q \in \Delta$ und jedes $\lambda \in [0,\ 1]$ gilt:

$$f\big((1-\lambda)P + \lambda Q\big) \leq (1-\lambda)f(P) + \lambda f(Q)\,,$$

wenn also der Graph von f über jeder Verbindungsstrecke zweier Punkte $P, Q \in \Delta$ unterhalb der Verbindungsstrecke der Punkte $\big(P, f(P)\big)$ und $\big(Q, f(Q)\big)$ liegt. Sie heißt *strikt konvex,* wenn dabei das Gleichheitszeichen nur für $\lambda = 0$ und $\lambda = 1$ gilt.
c) f heißt (strikt) *konkav,* wenn $-f$ (strikt) konvex ist.



Standardbeispiel einer konvexen Menge in $\mathbb{R}^n$ ist für uns die Menge

$$\Delta_n = \left\{(p_1, \ldots, p_n) \;\middle|\; p_i \geq 0 \;\text{ für alle } i \text{ und } \sum_{i=1}^{n} p_i = 1\right\}.$$

Sind $P = (p_1, \ldots, p_n)$ und $Q = (q_1, \ldots, q_n)$ zwei Punkte aus Δ_n, so ist

$$(1-\lambda)P + \lambda Q = \big((1-\lambda)p_1 + \lambda q_1, \ldots, (1-\lambda)p_n + \lambda q_n\big)\,.$$

Da alle p_i und q_i nichtnegativ sind, gilt für $\lambda \in [0,\ 1]$ dasselbe für die

Zahlen $(1-\lambda)p_i + \lambda q_i$, und

$$\sum_{i=1}^{n}\big((1-\lambda)p_i + \lambda q_i\big) = (1-\lambda)\sum_{i=1}^{n} p_i + \lambda \sum_{i=1}^{m} q_i = (1-\lambda) + \lambda = 1\,,$$

so daß auch $(1-\lambda)P + \lambda Q$ in Δ_n liegt.

Gerade bei der Definition einer konvexen Funktion erscheint es etwas seltsam, daß wir bei der Definition den Graphen nur über Strecken betrachten. Die Definition beschränkt sich auf diesen Fall, weil es sich um eine Eigenschaft handelt, die sich in vielen Fällen leicht nachprüfen läßt; tatsächlich gilt aber eine viel allgemeinere Aussage. Um sie auch für den Fall der strikten Konvexität zu formulieren, brauchen wir zunächst eine weitere Definition:

Definition: *a)* Eine Teilmenge $A \subset \mathbb{R}^n$ heißt r-dimensionaler affiner Unterraum von $\mathbb{R}^n$, wenn es einen Punkt $P_0 \in \mathbb{R}^n$ gibt, so daß die Verbindungsvektoren $\overrightarrow{P_0P}$ für die sämtlichen Punkte $P \in A$ einen r-dimensionalen Untervektorraum von $\mathbb{R}^n$ bilden.
b) m Punkte $P_1, \ldots, P_m \in \mathbb{R}^n$ sind *in allgemeiner Lage,* wenn es keinen $(m-2)$-dimensionalen affinen Unterraum $A \subseteq \mathbb{R}^n$ gibt, der alle diese Punkte enthält.

Zwei Punkte sind also genau dann in allgemeiner Lage, wenn sie verschieden sind; von dreien erwarten wir zusätzlich, daß sie nicht auf einer Geraden liegen, und von vieren, daß es keine Ebene gibt, die alle drei enthält.

Lemma: *a)* Eine Teilmenge $\Delta \subseteq \mathbb{R}^n$ ist genau dann konvex, wenn für jedes $m \in \mathbb{N}$ gilt: Sind $P_1, \ldots, P_m \in \Delta$ und ist $(\lambda_1, \ldots, \lambda_m) \in \Delta_m$, so liegt auch $\lambda_1 P_1 + \cdots + \lambda_m P_m$ in Δ.
b) Eine Funktion $f\colon \Delta \to \mathbb{R}$ auf einer konvexen Teilmenge $\Delta \subseteq \mathbb{R}^n$ ist genau dann konvex, wenn für je m Punkte $P_1, \ldots, P_m \in \Delta$ und jedes Tupel $(\lambda_1, \ldots, \lambda_m) \in \Delta_m$ gilt:

$$f(\lambda_1 P_1 + \cdots + \lambda_m P_m) \le \lambda_1 f(P_1) + \cdots + \lambda_m f(P_m)\,.$$

c) Eine Funktion $f\colon \Delta \to \mathbb{R}$ auf einer konvexen Teilmenge $\Delta \subseteq \mathbb{R}^n$ ist genau dann strikt konvex, wenn für je m Punkte $P_1, \ldots, P_m \in \Delta$ in

allgemeiner Lage und jedes Tupel $(\lambda_1, \ldots, \lambda_m) \in \Delta_m$ gilt:

$$f(\lambda_1 P_1 + \cdots + \lambda_m P_m) \leq \lambda_1 f(P_1) + \cdots + \lambda_m f(P_m)$$

mit Gleichheit genau dann, wenn ein $\lambda_i = 1$ ist und die übrigen λ_j verschwinden.

Beweis: Bei allen drei Behauptungen ist der Fall $m = 2$ gerade die Definition der (strikten) Konvexität; falls die Behauptung für *alle* $m \in \mathbb{N}$ gilt, gilt sie auch für $m = 2$, und die Rückrichtung ist bewiesen. Die andere Richtung ist trivial für $m = 1$, und für $m = 2$ ist sie jeweils die Definition. Wir müssen sie also nur für $m \geq 3$ beweisen, und dafür müssen wir die drei Behauptungen einzeln betrachten. Wir beweisen sie alle drei durch vollständige Induktion mit $m = 2$ als Induktionsanfang.

a) Wir haben m Punkte $P_1, \ldots, P_m \in \Delta$ und ein Tupel $(\lambda_1, \ldots, \lambda_m)$ aus Δ_m. Für $\lambda_m = 1$ verschwinden alle übrigen λ_j, und die Behauptung ist trivial; wir können uns also beschränken auf den Fall $\lambda_m \neq 1$. Dann können wir durch $1 - \lambda_m$ dividieren und das $(m-1)$-tupel

$$(\lambda_1^*, \ldots, \lambda_{m-1}^*) = \left(\frac{\lambda_1}{1-\lambda_m}, \ldots, \frac{\lambda_{m-1}}{1-\lambda_m} \right) \in \Delta_{m-1}$$

betrachten. Nach Induktionsannahme liegt der Punkt

$$P^* = \lambda_1^* P_1 + \cdots + \lambda_{m-1}^* P_{m-1}$$

daher in Δ, und nach Definition der Konvexität gilt dasselbe für $(1 - \lambda_m)P^* + \lambda_m P_m = \sum_{i=1}^m \lambda_i P_i$.

b) f sei konvex, $P_1, \ldots, P_m$ seien wieder Punkte aus Δ und $(\lambda_1, \ldots, \lambda_m)$ ein Tupel aus Δ_m, und P^* sei der oben definierte Punkt. Nach Induktionsannahme ist dann $f(P^*) \leq \lambda_1^* f(P_1) + \cdots + \lambda_{m-1}^* f(P_{m-1})$, und nach Definition der Konvexität von f ist außerdem

$$\begin{aligned} f\big((1-\lambda_m)P^* + \lambda_m P_m\big) &= f(\lambda_1 P_1 + \cdots + \lambda_m P_m) \\ \leq (1-\lambda_m) f(P^*) + \lambda_m f(P_m) &= \lambda_1 f(P_1) + \cdots + \lambda_m f(P_m)\,. \end{aligned}$$

c) Wir müssen nur noch zeigen, daß für Punkte in allgemeiner Lage Gleichheit nur gilt, wenn alle λ_i mit einer Ausnahme verschwinden und die Ausnahme damit gleich eins ist.

Falls $\lambda_m = 1$ ist, gibt es nichts mehr zu zeigen; wir können uns also auf den Fall $\lambda_m < 1$ beschränken. Dann können wir wie oben den Punkt P^* definieren.

Für Punkte $P_1, \ldots, P_m$ in allgemeiner Lage sind auch die beiden Punkte P^* und P_m in allgemeiner Lage, denn zwei Punkte sind genau dann in allgemeiner Lage, wenn sie verschieden sind, und wäre $P^* = P_m$, so wäre P_m eine Linearkombination von P_1 bis P_{m-1}, läge also im von diesen Punkten aufgespannten $(m-2)$-dimensionalen affinen Unterraum. Somit ist

$$\begin{aligned} f(\lambda_1 P_1 + \cdots + \lambda_m P_m) &= f\big((1-\lambda_m)P^* + \lambda_m P_m\big) \\ &= (1-\lambda_m) f(P^*) + \lambda_m f(P_n) \end{aligned}$$

genau dann, wenn $\lambda_m = 0$ oder $\lambda_m = 1$ ist. Den Fall $\lambda_m = 1$ haben wir bereits ausgeschlossen; also ist $\lambda_m = 0$. Dann aber folgt die Behauptung sofort aus der Induktionsannahme. ∎

Die Aussage unter *b)* wird auch als *Ungleichung von* JENSEN bezeichnet; er bewies sie in einem Vortrag vom 17. Januar 1905 vor der dänischen Mathematikergesellschaft, wobei er allerdings nur voraussetzte, daß die Funktion f auf einem reellen Intervall definiert ist und dort die Ungleichung $f(x) + f(y) \geq 2f\left(\frac{x+y}{2}\right)$ erfüllt, was auf den ersten Blick etwas schwächer aussieht als die hier betrachtete Definition der Konvexität, nach dem Resultat von JENSEN aber äquivalent dazu ist.

Der dänische Mathematiker Johan Ludwig William Valdemar Jensen (1859–1925) studierte ab 1876 an der Københavns Tekniske Skole unter anderem Mathematik, Physik und Chemie. Sein Interesse konzentrierte sich immer mehr auf die Mathematik; zwischen 1879 und 1925 veröffentlichte er rund vierzig wissenschaftliche Arbeiten. Er war allerdings nie an einer Universität tätig und war auch von der Ausbildung her im wesentlichen Autodidakt. Sein gesamtes Berufsleben arbeitete er als Telephoningenieur bei der dänischen Telephongesellschaft. Außer der heute nach ihm benannten Ungleichung bewies er unter anderem auch Sätze im Umkreis der RIEMANN-Vermutung.

Wenn wir die λ_i als Wahrscheinlichkeiten interpretieren, können wir *b)* und *c)* auch als Aussagen über Erwartungswerte interpretieren:

Lemma: Für eine diskrete Zufallsvariable X und eine $\left\{\begin{matrix}\text{konvexe}\\ \text{konkave}\end{matrix}\right\}$ Funktion f auf dem Wertebereich von X gilt: $\mathbb{E}\big(f(X)\big)\ \left\{\begin{matrix}\geq\\ \leq\end{matrix}\right\}\ f\big(\mathbb{E}(X)\big)$. Im Falle einer strikt $\left\{\begin{matrix}\text{konvexen}\\ \text{konkaven}\end{matrix}\right\}$ Funktion gilt Gleichheit genau dann, wenn X einen seiner Werte mit Wahrscheinlichkeit eins annimmt. ∎

Für mindestens zweimal stetig differenzierbare Funktionen läßt sich die Konvexität leicht anhand der zweiten Ableitung überprüfen. Im Fall einer Variablen haben wir einfach das

Lemma: *a)* Eine mindestens zweimal stetig differenzierbare Funktion $f\colon I \to \mathbb{R}$ auf einem Intervall $I \subseteq \mathbb{R}$ ist genau dann konvex, wenn ihre zweite Ableitung auf I keine negativen Werte annimmt; sie ist genau dann konkav, wenn f'' auf I keine positiven Werte annimmt.
b) Falls f'' im Innern von I nur positive Werte annimmt, ist f strikt konvex auf I; falls f'' dort nur negative Werte annimmt, ist f strikt konkav.

Beweis: a) Wir zeigen zunächst, daß im Falle der Konvexität die zweite Ableitung in ganz $(a,\, b)$ größer oder gleich null sein muß: Andernfalls gäbe es ein $x_0 \in (a,\, b)$ mit $f''(x_0) < 0$. Wir betrachten die Funktion $g(x) \underset{\text{def}}{=} f(x) - f'(x_0)(x - x_0)$. Als Summe von f und einer linearen Funktion ist g zweimal differenzierbar mit

$$g'(x_0) = f'(x_0) - f'(x_0) = 0 \quad \text{und} \quad g''(x_0) = f''(x_0) < 0\,.$$

Die Funktion $g'(x)$ ist also in einem hinreichend kleinen Intervall $(x_0 - h,\, x_0 + h)$ streng monoton fallend; sie ist daher positiv für $x < x_0$ und negativ für $x > x_0$. Somit ist g streng monoton wachsend für $x < x_0$ und streng monoton fallend für $x > x_0$; die Funktion g hat also bei x_0 ein lokales Maximum. Für ein $\varepsilon < h$ ist daher $g(x_0 \pm \varepsilon) < g(x_0)$ und damit ist auch

$$f(x_0) = g(x_0) > \frac{1}{2}g(x_0 - \varepsilon) + \frac{1}{2}g(x_0 + \varepsilon) = \frac{1}{2}f(x_0 - \varepsilon) + \frac{1}{2}f(x_0 + \varepsilon)\,.$$

Dies widerspricht aber der Konvexitätsbedingung für $x_{1/2} = x_0 \pm \varepsilon$ und $\lambda = \frac{1}{2}$. Somit muß $f''(x)$ in ganz $(a,\, b)$ größer oder gleich null sein.

Umgekehrt sei $f''(x) \geq 0$ für alle $x \in (a, b)$; wir müssen zeigen, daß f dann konvex ist. Seien also $x_1 < x_2$ zwei beliebige Punkte aus (a, b) und $x = (1 - \lambda)x_1 + \lambda x_2$ mit $\lambda \in (0, 1)$. Nach dem Mittelwertsatz gibt es Punkte $\xi_1 \in (x_1, x)$ und $\xi_2 \in (x, x_2)$, so daß

$$f'(\xi_1) = \frac{f(x) - f(x_1)}{x - x_1} = \frac{f(x) - f(x_1)}{\lambda(x_2 - x_1)} \quad \text{und}$$
$$f'(\xi_2) = \frac{f(x_2) - f(x)}{x_2 - x} = \frac{f(x_2) - f(x)}{(1 - \lambda)(x_2 - x_1)}$$

ist. Da f'' nirgends negativ wird, ist f' monoton wachsend und damit insbesondere $f'(\xi_1) \leq f'(\xi_2)$. Diese Ungleichung können wir auch schreiben als

$$\frac{f(x) - f(x_1)}{\lambda(x_2 - x_1)} \leq \frac{f(x_2) - f(x)}{(1 - \lambda)(x_2 - x_1)} ,$$

und da $x_1 < x_2$ ist, folgt daraus

$$\frac{f(x) - f(x_1)}{\lambda} \leq \frac{f(x_2) - f(x)}{1 - \lambda} .$$

Für $\lambda \in (0, 1)$ ändert sich nichts an dieser Ungleichung, wenn wir mit $\lambda(1 - \lambda)$ multiplizieren; dies führt auf

$$(1 - \lambda)f(x) - (1 - \lambda)f(x_1) \leq \lambda f(x_2) - \lambda f(x)$$

und damit die gewünschte Ungleichung

$$f(x) \leq (1 - \lambda)f(x_1) + \lambda f(x_2) ,$$

die die Konvexität von f ausdrückt. Damit ist die Behauptung für konvexe Funktionen bewiesen.

Für konkave Funktionen folgt sie einfach daraus, daß $-f$ für eine konkave Funktion f konvex ist.

■

Korollar: Die Funktion $f(x) = -x \log x$ ist über dem Intervall $[0, 1]$ konkav.

Beweis: $f'(x) = -x \cdot \frac{1}{x} - \log x = -1 - \log x$ hat als Ableitung die Funktion $f''(x) = -1/x$, die im Intervallinnern überall negativ ist.

■

Damit gilt insbesondere für zwei beliebige Zahlen $p_1, p_2 \in [0,\ 1]$, daß

$$-\frac{p_1+p_2}{2}\log_2\frac{p_1+p_2}{2} \geq \frac{1}{2}(-p_1\log_2 p_1)+\frac{1}{2}(-p_2\log_2 p_2)$$

oder

$$-2\cdot\frac{p_1+p_2}{2}\log_2\frac{p_1+p_2}{2} \geq -p_1\log_2 p_1 - p_2\log_2 p_2\,.$$

Ersetzt man also im Ausdruck

$$H(p_1,\ldots,p_n) = -\sum_{i=1}^{m} p_i\log_2 p_i$$

irgendwelche zwei *verschiedene* Wahrscheinlichkeiten p_i und p_j durch ihren gemeinsamen Mittelwert $\frac{1}{2}(p_i+p_j)$, so wird die Entropie größer. Damit folgt fast sofort

Satz: Für m Zahlen $p_1,\ldots,p_m \in [0,1]$ mit $\sum\limits_{i=1}^{m} p_i = 1$ gilt stets

$$0 \leq H(p_1,\ldots,p_m) = -\sum_{i=1}^{m} p_i\log_a p_i \leq \log m\,;$$

dabei steht rechts genau dann ein Gleichheitszeichen, wenn alle p_i gleich $1/m$ sind und links steht genau dann eines, wenn alle p_i mit einer Ausnahme verschwinden.

Beweis: Da H eine stetige Funktion auf der kompakten Menge Δ_m ist, nimmt sie sowohl ihr Maximum als auch ihr Minimum an. Wie wir gerade gesehen haben, kann es im Maximum keine zwei echt verschiedenen p_i geben, also müssen alle $p_i = \frac{1}{m}$ sein und das Maximum ist

$$H\big(\tfrac{1}{m},\ldots,\tfrac{1}{m}\big) = m\cdot\big(-\tfrac{1}{m}\log_2\tfrac{1}{m}\big) = -\log_2\tfrac{1}{m} = \log_2 m\,.$$

Umgekehrt ist $-p_i\log_2 p_i \geq 0$ für alle $p_i \in [0,\ 1]$ mit Gleichheit genau dann, wenn $p_i = 0$ oder $p_i = 1$ ist. Eine Summe Null entsteht somit genau dann, wenn alle $p_i \in \{0,1\}$ sind, d.h. wenn genau ein $p_i = 1$ ist und der Rest verschwindet.

■

§3: Ein Beispiel

Um ein Gefühl für den SHANNONschen Informationsbegriff zu bekommen, wollen wir ein bekanntes (1984 von ALFRÉD RÉNYI (1921–1970) beschriebenes) Ratespiel informationstheoretisch betrachten: Gegeben sind zwölf gleich aussehende Kugeln, von denen mindestens elf dasselbe Gewicht haben, sowie eine Balkenwaage. Man finde mit höchstens dreimaligem Wiegen heraus, ob es eine Kugel mit abweichendem Gewicht gibt, welche dies ist und ob sie leichter oder schwerer als der Rest ist.

Auf diese Frage gibt es 25 mögliche Antworten, die wir auf Grund unseres Informationsstands als gleich wahrscheinlich betrachten müssen; die korrekte Antwort hat somit einen Informationsgehalt von $\log_2 25$ Bit. Beim Wiegen erhalten wir eines von drei möglichen Ergebnissen (linke Seite schwerer, rechte Seite schwerer, beide Seiten gleich schwer); falls es uns gelingt, die zu vergleichenden Kugeln so auszuwählen, daß alle drei Ergebnisse gleich wahrscheinlich sind, bekommen wir eine Information von $\log_2 3$ Bit pro Wiegen. Bei dreimaligem Wiegen wären das $3 \cdot \log_2 3 = \log_2 3^3 = \log_2 27$ Bit, was mehr ist als $\log_2 25$ Bit. Von daher spricht also nichts gegen die Lösbarkeit der Aufgabe, allerdings haben wir auch nicht viel Spielraum und müssen daher bei jedem Wiegen unbedingt darauf achten, daß die drei möglichen Resultate mit zumindest ungefähr gleicher Wahrscheinlichkeit auftreten.

Damit verbietet sich insbesondere der naheliegende Ansatz, zunächst zwei Sechsergruppen von Kugeln miteinander zu vergleichen: Da die Waage nur im Fall, daß alle Kugeln das gleiche Gewicht haben. im Gleichgewicht ist, tritt hier einer der drei Fälle nur mit einer Wahrscheinlichkeit von $1/25$ auf, die beiden anderen jeweils mit $12/25$, so daß wir nur eine Information von

$$-\frac{1}{25}\log_2\frac{1}{25} - 2\cdot\frac{12}{25}\log_2\frac{12}{25} \approx 1{,}202$$

Bit bekommen, was um etwa 0,383 Bit unter $\log_2 3 \approx 1{,}585$ liegt. Da $\log_2 27 - \log_2 25$ nur ungefähr 0,111 ist, kämen wir selbst, wenn wir bei den nächsten beiden Versuchen jeweils die vollen $\log_2 3$ Bit realisieren könnten, nicht auf die notwendige Information von $\log_2 25$ Bit,

Das Hauptproblem bei den Sechsergruppen besteht darin, daß die Wahrscheinlichkeit gleich schwerer Seiten so gering ist. Diese Wahrscheinlichkeit können wir erhöhen, indem wir beim Wiegen nicht alle zwölf Kugeln in eine der beiden Waagschalten legen: Legen wir r Kugeln beiseite, wobei $12 - r$ natürlich eine gerade Zahl sein muß, so erhalten wir ein Gleichgewicht, wenn entweder alle zwölf Kugeln gleich schwer sind oder wenn eine der r zur Seite gelegten Kugeln schwerer oder leichter als der Rest ist, also in $2r + 1$ Fällen. Diese Zahl sollte möglichst nahe bei $\frac{25}{3} = 8\frac{1}{3}$ liegen; somit bietet sich $r = 4$ an, d.h. wir bilden drei Viergruppen.

Wir numerieren also die Kugeln von 1 bis 12 und vergleichen die Kugeln 1 bis 4 mit 5 bis 8. Wie wir uns gerade überlegt haben, erhalten wir in neun der 25 Fälle das Ergebnis *gleich schwer.* Die linke Seite mit den Kugeln 1 bis 4 ist genau dann schwerer, wenn entweder eine dieser vier Kugeln schwerer ist als die anderen oder wenn eine der Kugeln 5 bis 9 leichter ist als die anderen, also in jeweils acht Fällen. In den verbleibenden acht Fällen ist die rechte Seite schwerer; wir haben also drei Ergebnisse mit Wahrscheinlichkeiten $9/25$ und zweimal $8/25$. Die Entropie des Wiegevorgangs beträgt somit

$$-\frac{9}{25}\log_2\frac{9}{25} - 2\cdot\frac{8}{25}\log_2\frac{8}{25} \approx 1{,}583\,,$$

was nur um etwa 0,002 unter der maximal möglichen Information von $\log_2 3$ Bit liegt. (Diese können wir hier natürlich nicht erreichen, da 25 nicht durch drei teilbar ist.)

Wenn beide Seiten gleich schwer waren, wissen wir nicht nur, daß die gesuchte Kugel, so sie existiert, eine Nummer zwischen neun und zwölf hat, sondern wir wissen auch, daß die Kugeln eins bis acht allesamt das „übliche“ Gewicht haben. Wir haben somit „Referenzkugeln“, mit denen wir entscheiden können, ob eine gegebene Kugel leichter oder schwerer ist als der Rest.

Insgesamt haben wir neun mögliche Fälle (alle Kugeln gleich schwer, eine der Kugeln neun bis zwölf leichter *bzw.* schwerer); wir sollten so wiegen, daß jedes der drei möglichen Ergebnisse in drei der neun Fälle eintritt.

Dazu können wir beispielsweise die zwölfte Kugel auszeichnen und als eine Gruppe von drei Fällen den nehmen, daß entweder alle Kugeln gleich schwer sind oder aber die zwölfte das falsche Gewicht hat. In diesen drei Fällen haben also die Kugeln neun bis elf das richtige Gewicht.

Dies können wir entscheiden, in dem wir sie mit drei Referenzkugeln vergleichen, etwa den Kugeln eins bis drei; in den drei betrachteten Fällen sind beide Seiten der Waage gleich schwer.

Falls die Kugeln eins bis drei schwerer sind als neun bis elf, ist eine der letzteren leichter als der Rest, wofür es drei Fälle gibt; in den verbleibenden drei Fällen, wenn eine der Kugeln neun bis elf schwerer ist als der Rest, sind auch die drei Kugeln zusammen schwerer als eins bis drei. Hier erhalten wir also die maximal mögliche Information von $\log_2 3$ Bit.

Falls die Waage im Gleichgewicht war, ist klar, wie wir weiter wiegen: Wir vergleichen Kugel zwölf mit irgendeiner anderen Kugel und erfahren, ob sie schwerer, leichter oder gleich schwer wie die anderen Kugeln ist; in diesem Fall liefert uns also auch das dritte Wiegen eine Information von $\log_2 3$ Bit.

In den beiden anderen Fällen wissen wir entweder, daß eine der drei Kugeln neun bis elf leichter ist als der Rest oder daß sie schwerer ist; wir müssen nur noch herausfinden, um welche der drei Kugeln es sich handelt. Dazu können wir beispielsweise die Kugeln neun und zehn miteinander vergleichen: Sind sie gleich schwer, so hat elf das abweichende Gewicht, andernfalls ist es im ersten Fall die leichtere, im zweiten die schwerere der beiden Kugeln. Hier erhalten wir also beim Wiegen wieder die maximal mögliche Information von $\log_2 3$ Bit.

Damit sind alle Fälle abgehandelt, bei denen die Waage beim ersten Einsatz ausbalanciert war; bleiben noch die, daß eine der beiden Seiten schwerer war.

Angenommen, die Kugeln von eins bis vier sind schwerer als die von fünf bis acht. Dann ist entweder eine der Kugeln eins bis vier zu schwer ist oder eine der Kugeln fünf bis acht zu leicht. Da wir in diesem Fall acht gleich wahrscheinliche Möglichkeiten haben und acht nicht

durch drei teilbar ist, können wir beim Wiegen keine Information von $\log_2 3$ Bit bekommen; am meisten Information erhalten wir, wenn zwei der möglichen Ergebnisse in jeweils drei Fällen auftreten und das dritte in zweien. Ein solches Experiment, falls wir es realisieren können, liefert eine Information von

$$-2 \cdot \frac{3}{8} \log_2 \frac{3}{8} - \frac{1}{4} \log_2 \frac{1}{4} = -\frac{3}{4}(\log_2 3 - \log_2 8) + \frac{1}{4} \log_2 4$$
$$= -\frac{3}{4} \log_2 3 + \frac{9}{4} + \frac{2}{4} = \frac{11}{4} - \frac{3}{4} \log_2 3 \approx 1{,}561$$

Bit, was um etwa 0,024 kleiner ist als $\log_2 3 \approx 1{,}585$.

Im Gegensatz zur obigen Situation gibt es hier für jede Kugel nur noch zwei Möglichkeiten: Wenn ihr Gewicht von dem der restlichen Kugeln abweicht, ist es im Falle der Kugeln eins bis vier notwendigerweise zu schwer, bei den vier anderen notwendigerweise zu leicht. Wir können deshalb keine Gruppe aus zwei Fällen konstruieren, indem wir nur *eine* Kugel betrachten; wir brauchen mindestens zwei.

Versuchen wir also, die beiden Fälle *Kugel eins zu schwer* und *Kugel zwei zu schwer* zu einer Fallgruppe zusammenzufassen. Die restlichen sechs Fälle müssen wir zu zwei Dreiergruppen zusammenfassen; aus Symmetriegründen sollte jede von diesen einen der beiden Fälle *Kugel drei zu schwer* und *Kugel vier zu schwer* enthalten sowie zwei Fälle mit zu leichten Kugeln.

Damit ist klar, wie wir weiter vorgehen können: Wir legen beispielsweise die Kugeln drei, fünf, sechs in die linke und vier, sieben, acht in die rechte Waagschale. Die linke Waagschale geht nach unten, wenn drei schwerer oder sieben oder acht leichter ist, die rechte, wenn vier schwerer oder fünf oder sechs leichter ist. In den verbleibenden Fällen, daß eins oder zwei schwerer ist, halten sich beide Seiten die Waage, und wir müssen wir nur noch eins und zwei vergleichen; die schwerere der beiden Kugeln ist die abweichende, und sie ist schwerer als der Rest. Der Informationsgehalts dieses Vergleichs ist somit nur ein Bit.

In den anderen Fällen vergleichen wir jeweils die beiden möglicherweise zu leichten Kugeln. Ist eine davon tatsächlich leichter als die andere, ist

sie die Lösung; andernfalls haben beide dasselbe Gewicht und die potentiell schwerere Kugel ist wirklich schwerer als der Rest. Hier bekommen wir also wieder $\log_2 3$ Bit Information.

Bleibt noch der Fall, daß die Kugeln von eins bis vier *leichter* sind als die von fünf bis acht; hier können wir natürlich vorgehen wie eben, nur daß die Begriffe *leichter* und *schwerer* vertauscht werden müssen.

Beim ersten Wiegen erhalten wir somit eine Information von

$$-\frac{16}{25}\log_2\frac{8}{25}-\frac{9}{25}\log_2\frac{9}{25} = \frac{-16}{25}(\log_2 8-\log_2 25)-\frac{9}{25}(\log_2 9-\log_2 25)$$
$$= \log_2 25 - \frac{48}{25} - \frac{18}{25}\log_2 3$$

Bit. In den 9/25 aller Fälle, in denen die Waage im Gleichgewicht ist, konnten wir beim zweiten und dritten Wiegen jeweils die maximal mögliche Information von $\log_2 3$ Bit realisieren, insgesamt also $2\log_2 3$ Bit. In den übrigen Fällen erhalten wir beim zweiten Wiegen nur eine Information von $\frac{11}{4} - \frac{3}{4}\log_2 3$ Bit und beim dritten erhalten wir in einem Viertel der Fälle nur ein Bit, ansonsten $\log_2 3$ Bit. Im Mittel bekommen wir somit genau die benötigte Information von

$$\left(\log_2 25 - \frac{48}{25} - \frac{18}{25}\log_2 3\right)$$
$$+\frac{9}{25}\cdot 2\log_2 3 + \frac{16}{25}\left(\frac{11}{4} - \frac{3}{4}\log_2 3 + \frac{1}{4} + \frac{3}{4}\log_2 3\right)$$
$$= \log_2 25 - \frac{48}{25} + \frac{16}{25}\cdot\frac{12}{4} = \log_2 25 \text{ Bit}\,.$$

Bei n Kugeln, von denen genau eine entweder schwerer oder leichter als die übrigen ist, haben wir offensichtlich keine Chance, das Problem mit r-maligem Wiegen zu lösen, wenn $\log_2(2n+1) > r\log_2 3$ ist oder, äquivalent, $2n+1 > 3^r$; schließlich können wir beim Wiegen nie eine größere Information als $\log_2 3$ Bit erhalten, und zumindest in einigen Fällen erhalten wir zwangsläufig weniger Information. Man kann sich fragen, ob wir im Falle $\log_2(2n+1) \le r\log_2 3$ *immer* eine Strategie finden können, bei der wir mit r maligem Wiegen auskommen. In diesem Fall sollte es also insbesondere möglich sein, mit dreimaligem Wiegen

nicht nur das Problem mit zwölf Kugeln zu lösen, sondern sogar das mit dreizehn.

Hier gibt es 27 Möglichkeiten; um beim ersten Wiegen die maximal mögliche Information zu bekommen, sollten wir ein Experiment durchführen, bei dem jede der drei Alternativen genau neun Mal eintritt. Beim ersten Wiegen gibt es aber im wesentlichen nur einen Parameter, den wir beeinflussen können: Wir wählen irgendeine Zahl $m \leq 6$ und legen in beide Waagschalen jeweils m Kugeln. In je $2m$ Fällen geht dann die linke oder rechte Waagschale nach unten; in den verbleibenden $27 - 4m$ Fällen sind sie ausbalanciert. Da sich neun nicht in der Form $9 = 2m$ schreiben läßt, können wir somit schon beim ersten Wiegen nicht die erforderliche Information von $\log_2 3$ Bit erreichen.

§4: Die Entropierate stochastischer Prozesse

a) Stochastische Prozesse

Im vorigen Paragraphen waren wir ausgegangen von einer Folge unabhängiger Zufallsvariablen. Wenn wir eine Quelle modellieren wollen, die beispielsweise deutsche Texte produziert, ist das sicherlich eine unrealistische Annahme: E ist der häufigste Buchstabe des Alphabets, aber hinter einem E kommt fast nie ein weiteres E, und auch hinter einem C kommt nur selten ein E; viel wahrscheinlicher sind hier die (ansonsten eher nicht so häufigen) Buchstaben H und K.

Um zu realistischeren Modellen zu kommen, müssen wir daher auch Abhängigkeiten zwischen den Wahrscheinlichkeitsverteilungen der einzelnen Zufallsvariablen zulassen und damit auch allgemeinere stochastische Prozesse zulassen. Darunter verstehen wir hier einfach Folgen $X_1, X_2, \ldots$ von Zufallsvariablen mit Werten in einem festen endlichen Alphabet A.

Bei Prozessen, die natürliche Sprachen beschreiben, sollte die Wahrscheinlichkeit, mit der ein gegebenes Wort vorkommt, nicht davon abhängen, ob wir den Anfang oder das Ende des Textes betrachten; solche Prozesse bezeichnen wir als *stationär:*

Definition: Ein stochastischer Prozess $(X_k)_{k\in\mathbb{N}}$ heißt *stationär,* wenn für alle $n \in \mathbb{N}$, alle $(x_1, \ldots, x_n) \in A^n$ und alle $m \in \mathbb{N}$ gilt: Die Wahrscheinlichkeit des Ereignisses $X_{m+1} = x_1, X_{m+2} = x_2, \ldots, X_{m+n} = x_n$ ist gleich der des Ereignisses $X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n$.

Stochastische Prozesse, die reale Phänomene beschreiben, haben oft sehr komplizierte Wahrscheinlichkeitsverteilungen; insbesondere wird der Wert von X_n oftmals von vielen, wenn nicht gar allen Werten der Vorgänger abhängen. Als einfache Idealisierung, die immerhin noch etwas realistischer ist als eine Folge unabhängiger Zufallsvariablen, sind MARKOV-Ketten ein beliebtes Modell. Hierbei handelt es sich um stochastische Prozesse ohne Gedächtnis, d.h. die Wahrscheinlichkeit, mit der die Zufallsvariable X_n einen Wert produziert, hängt nur ab vom Wert des unmittelbaren Vorgängers X_{n-1}. Formal:

Definition: *a)* Ein stochastischer Prozess $X_1, X_2, \ldots$ heißt MARKOV-Prozess oder MARKOV-Kette, wenn für alle $n \in \mathbb{N}$ gilt

$$p(X_{n+1} = x_{n+1} \mid X_1 = x_1, \ldots, X_n = x_n) = p(X_{n+1} = x_{n+1} \mid X_n = x_n)\,.$$

b) Eine MARKOV-Kette heißt *zeitinvariant,* wenn die bedingte Wahrscheinlichkeit $p(X_{n+1} = y \mid X_n = x)$ nicht von n abhängt. Ist $A = \{a_1, \ldots, a_m\}$, so setzen wir $p_{ij} = p(X_{n+1} = a_j \mid X_n = a_i)$ und bezeichnen die $m \times m$-Matrix M mit Einträgen p_{ij} als die *Übergangsmatrix* des Prozesses.

c) Eine MARKOV-Kette heißt *irreduzibel,* wenn es für je zwei Buchstaben $a, b \in A$ und jede natürliche Zahl n ein $r \in \mathbb{N}$ gibt, so daß $p(X_{n+r} = b \mid X_n = a) > 0$ ist.

Der russische Mathematiker ANDREĬ ANDREEVIČ MARKOV (Андрей Андреевич Марков, 1856–1922) studierte in Sankt Petersburg, wo er später auch Professor wurde. Er beschäftigte sich zunächst hauptsächlich mit Zahlentheorie und Analysis; erst später folgen die wahrscheinlichkeitstheoretischen Arbeiten, für die er heute vor allem bekannt ist. Der Name Марков wird in lateinischen Buchstaben verschieden transkribiert; MARKOVs französische Arbeiten erschienen mit der Schreibweise MARKOFF; nach den klassischen deutschen Transkriptionsregeln müßte man MARKOW schreiben. Die Schreibweise MARKOV entspricht den eng-

lischen Regeln und scheint sich mittlerweile in der Mathematik ziemlich durchgesetzt zu haben.

Bei einer irreduziblen MARKOV-Kette gibt es also keinen Buchstaben, dessen Auftreten das künftige Auftreten irgendeines anderen Buchstaben verhindert.

Wir werden im folgenden, soweit nicht explizit etwas anderes gesagt ist, stets annehmen, daß unsere MARKOV-Ketten zeitinvariant sind. In diesem Fall können wir die Wahrscheinlichkeitsverteilungen aller Zufallsvariablen aus der von X_1 und der Übergangsmatrix berechnen: Ist allgemein $p_i^{(n)}$ die Wahrscheinlichkeit, mit der X_n dem Wert a_i annimmt, so ist

$$\begin{aligned} &p(X_n = a_{i_0}, X_{n+1} = a_{i_1}, \ldots, X_{n+r} = a_{i_r}) \\ &= p(X_n = a_{i_0}) \prod_{\ell=1}^{r} p(X_{n+\ell} = a_{i_\ell} \mid X_{n+\ell-1} = a_{\ell-1}) \,. \end{aligned}$$

Für $r = 1$ wird das zu

$$p(X_n = a_i, X_{b+1} = a_j) = p(X_n = a_i) p(X_{n+1} = a_j \mid X_n = a_i) = p_i^{(n)} p_{ij} \,,$$

was wir auch einfacher mit Matrizen und Vektoren formulieren können: Ist $\mathfrak{p}^{(n)} = (p_1^{(n)}, \ldots, p_m^{(n)})^T$ der Spaltenvektor der Wahrscheinlichkeitsverteilung zu X_n, so ist $\mathfrak{p}^{(n+1)} = M^T \mathfrak{p}^{(n)}$ und damit $\mathfrak{p}^{(n)} = \left(M^T\right)^{n-1} \mathfrak{p}^{(1)}$.

Somit bestimmen $\mathfrak{p}^{(1)}$ und die Übergangsmatrix M die Wahrscheinlichkeitsverteilungen aller X_n und erlauben damit auch die Berechnung der Wahrscheinlichkeiten aller Teiltupel, die von der MARKOV-Kette produziert werden.

MARKOV-Ketten sind noch kein realistisches Modell für eine natürliche Sprache: Im Deutschen folgen beispielsweise auf ein C vorzugsweise die Buchstaben H und K; falls vor dem C aber ein S steht, sinkt die Wahrscheinlichkeit für ein K dramatisch. Trotzdem liefern MARKOV-Ketten ein deutlich besseres Modell für die deutsche Sprache als unabhängige Zufallsvariablen.

Wenn es uns darum geht, das Ergebnis eines stochastischen Prozesses zu kodieren, interessiert für lange Folgen vor allem die *mittlere* Entropie

oder *Entropierate*

$$H \underset{\text{def}}{=} \lim_{n\to\infty} \frac{H(X_1,\ldots,X_n)}{n}$$

– sofern dieser Grenzwert existiert. Es ist nicht schwer, Beispiele zu finden, in denen er nicht existiert; bei den Prozessen, die uns interessieren, werden wir aber keine Probleme haben.

b) Wechselseitige Information

Bevor wir die mittlere Entropie einer MARKOV-Kette berechnen können, brauchen wir noch einige Vorbereitungen über die Entropie voneinander abhängiger Zufallsvariablen.

Wir betrachten daher zwei Zufallsvariablen X, Y mit Werten in nicht notwendigerweise übereinstimmenden Alphabeten A, B. Die gemeinsame Entropie der beiden ist einfach die Entropie der Zufallsvariablen $X \times Y$ mit Werten in $A \times B$, also

$$H(X,Y) = -\sum_{a\in A}\sum_{b\in B} p(X=a, Y=b)\log_2 p(X=a, Y=b)\,.$$

Die Abhängigkeit zwischen X und Y wird beschrieben durch die bedingten Wahrscheinlichkeiten; entsprechend dazu definieren wir

Definition: Die *bedingte Entropie* zweier Zufallsvariablen X, Y ist

$$H(Y|X) = -\sum_{a\in A}\sum_{b\in B} p(X=a, Y=b)\log_2 p(Y=b|X=a)\,;$$

für n Zufallsvariablen ist entsprechend

$$H(X_n|X_{n-1},\ldots,X_1) =$$

$$-\sum_{a\in A_1\times\cdots\times A_n} p(X_1=a_1,\ldots,X_n=a_n)\log_2 p(X_n=a_n|X_{n-1}=a_{n-1},\ldots,X_1=a_1)\,.$$

Um Platz zu sparen werden wir künftig meist kurz $p(a_1,\ldots,a_n)$ und $p(a_n \mid a_{n-1},\ldots,a_1)$ schreiben.

Für die bedingte Entropie gilt die folgende

Kettenregel: $H(X,Y) = H(X) + H(Y|X)$ und allgemein

$$H(X_1,\dots,X_n) = \sum_{i=1}^{n} H(X_i|X_{i-1},\dots,X_1)\,.$$

Beweis: Der Übersichtlichkeit halber sei zunächst der Fall $n = 2$ behandelt; hier ist

$$\begin{aligned}
H(X,Y) &= -\sum_{a\in A}\sum_{b\in B} p(X=a,Y=b)\log_2 p(X=a,Y=b)\\
&= -\sum_{a\in A}\sum_{b\in B} p(X=a,Y=b)\log_2\big(p(X=a)p(Y=b|X=a)\big)\\
&= -\sum_{a\in A}\left(\sum_{b\in B} p(X=a,Y=b)\right)\log_2 p(X=a)\\
&\qquad -\sum_{a\in A}\sum_{b\in B} p(X=a,Y=b)\log_2 p(Y=b|X=a)\\
&= -\sum_{a\in A} p(X=a)\log_2 p(X=a) + H(Y|X)\\
&= H(X) + H(Y|X)\,.
\end{aligned}$$

Der allgemeine Fall geht genauso: $H(X_1,\dots,X_n)$ ist nach Definition gleich

$$\begin{aligned}
&-\sum_{a\in A_1\times\cdots\times A_n} p(a_1,\dots,a_n)\log_2 p(a_1,\dots,a_n)\\
&= -\sum_{a\in A_1\times\cdots\times A_n} p(a_1,\dots,a_n)\log_2\prod_{i=1}^{n} p(a_i|a_{i-1},\dots,a_1)\\
&= -\sum_{a\in A_1\times\cdots\times A_n}\sum_{i=1}^{n} p(a_1,\dots,a_n)\log_2 p(a_i|a_{i-1},\dots,a_1)\\
&= -\sum_{i=1}^{n}\sum_{a\in A_1\times\cdots\times A_n} p(a_1,\dots,a_n)\log_2 p(a_i|a_{i-1},\dots,a_1)
\end{aligned}$$

$$= -\sum_{i=1}^{n} \sum_{a \in A_1 \times \cdots \times A_i} p(a_1, \ldots, a_n) \log_2 p(a_i | a_{i-1}, \ldots, a_1)$$

$$= \sum_{i=1}^{n} H(X_i | X_{i-1}, \ldots, X_1) ,$$

denn $\sum_{(a_{i+1}, \ldots, a_n) \in A_{i+1} \times \cdots \times A_n} p(a_1, \ldots, a_n) = p(a_1, \ldots, a_i).$ ∎

Die bedingte Entropie ist nicht das einzige Maß für die Information, die eine Zufallsvariable über eine andere gibt; um noch ein anderes zu definieren, betrachten wir zunächst den Fall zweier Zufallsvariablen X, Y mit Werten im gleichen Alphabet A. Diese unterscheiden sich nur in den Wahrscheinlichkeitsverteilungen: X nehme den Wert a an mit Wahrscheinlichkeit $p(a)$, Y mit Wahrscheinlichkeit $q(a)$.

Definition: Die KULLBACK-LEIBLER-Distanz zwischen X und Y oder p und q ist

$$D(X \| Y) = D(p \| q) = \sum_{a \in A} p(a) \log_2 \frac{p(a)}{q(a)} .$$

Man beachte, daß die KULLBACK-LEIBLER-Distanz trotz des Namens *Distanz* keine Metrik ist: Im allgemeinen ist $D(p\|q) \neq D(q\|p)$. Andere Namen für $D(p\|q)$ sind *relative Entropie* oder KULLBACK-LEIBLER-*Divergenz.*

$D(p\|q)$ ist allerdings, wie es sich für eine Distanz gehört, nie negativ, denn wegen der Konkavität des Logarithmus ist

$$\sum_{a \in A} p(a) \log_2 \frac{p(a)}{q(a)} = -\sum_{a \in A} p(a) \log_2 \frac{q(a)}{p(a)}$$

$$\geq -\log_2 \left(\sum_{a \in A} p(a) \frac{q(a)}{p(a)} \right) = -\log_2 \sum_{a \in A} q(a) = -\log_2 1 = 0 .$$

SOLOMON KULLBACK (1907–1994) studierte Mathematik am City College of New York und arbeitete zunächst als Lehrer. Schon 1930 wechselte er zum Signals Intelligence Service in Washington, D.C., wo er bei WILLIAM FRIEDMAN Kryptologie lernte. Daneben promovierte er 1934 an der George Washington University mit einer Arbeit aus dem Gebiet der Statistik. Während des zweiten Weltkriegs beschäftigte er sich mit dem Knacken deutscher und japanischer Codes; nach Gründung der *National Security Agency* im Jahr 1952 leitete er dort die Forschung und Entwicklung. Nach seiner Pensionierung 1962 arbeitete er als Professor für Statistik an der George Washington University.

RICHARD ARTHUR LEIBLER (1914–2003) studierte Mathematik an der Northwestern University; nachdem er dort seinen Master erhalten hatte, promovierte er an der University of Illinois mit einer Arbeit über nichtlineare Differentialgleichungen. Nach einer kurzen Tätigkeit als Lehrer wechselte er zur Navy, die ihn im zweiten Weltkrieg im Pazifik einsetzte. 1953 kam er zur *National Security Agency,* wo er zunächst in der Forschungs- und Entwicklungsabteilung arbeitete; 1957 wurde er Leiter der Mathematischen Forschungsabteilung. 1958 wechselte er zur Kommunikationsabteilung des Instituts für Verteidigungsuntersuchungen in Princeton, deren Direktor er 1962 wurde. Von 1977 bis zu seiner Pensionierung 1980 arbeitete er wieder bei der NSA.

Um aus der KULLBACK-LEIBLER-Distanz ein Maß für die Abhängigkeit zweier beliebiger Zufallsvariablen X mit Werten in A und Y mit Werten in B zu machen, betrachten wir zwei Wahrscheinlichkeitsverteilungen auf $A \times B$, nämlich einmal die gemeinsame Verteilung von X und Y, zum anderen die Verteilung, die wir hätten, wenn X und Y unabhängig wären, wenn also das Ereignis $(X = a, Y = b)$ die Wahrscheinlichkeit $p(X = a)p(Y = b)$ hätte. Die Distanz zwischen diesen beiden Verteilungen gibt uns ein Maß für die Abhängigkeit der beiden Zufallsvariablen:

Definition: Die *wechselseitige Information* der Zufallsvariablen X, Y ist $I(X;Y) \underset{\text{def}}{=} \sum_{a\in A} \sum_{b\in B} p(X = a, Y = b) \log_2 \dfrac{p(X = a, Y = b)}{p(X = a)p(Y = b)}$.

Als spezieller Fall einer KULLBACK-LEIBLER-Distanz ist die wechselseitige Information natürlich immer größer oder gleich Null.

Da $p(X = a, Y = b) = p(Y = b)p(X = a|Y = b)$ ist, können wir sie auch schreiben als

$$\begin{aligned}
I(X;Y) &= \sum_{a\in A}\sum_{b\in B} p(X=a,Y=b)\frac{\log_2 p(Y=b)p(X=a|Y=b)}{p(X=a)p(Y=b)} \\
&= \sum_{a\in A}\sum_{b\in B} p(X=a,Y=b)\frac{\log_2 p(X=a|Y=b)}{p(X=a} \\
&= \sum_{a\in A}\sum_{b\in B} p(X=a,Y=b)\log_2 p(X=a|Y=b) \\
&\qquad - \sum_{a\in A}\left(\sum_{b\in B} p(X=a,Y=b)\right)\log_2(X=a) \\
&= \sum_{a\in A}\sum_{b\in B} p(X=a,Y=b)\log_2 p(X=a|Y=b) \\
&\qquad - \sum_{a\in A} p(X=a)\log_2(X=a) \\
&= H(X) - H(X|Y)\,.
\end{aligned}$$

Somit ist $I(X;Y) = H(X) - H(X|Y)$ gerade die Differenz zwischen der Entropie von X und der bedingten Entropie von X bei Kenntnis von Y, was den Begriff *wechselseitige Information* besser erklärt als die Formel aus der Definition. Die Nichtnegativität von $I(X;Y)$ beschreibt die Tatsache, daß die Entropie einer Zufallsvariable durch Zusatzinformation höchstens kleiner werden kann.

Auch die durch das Wort *wechselseitig* implizierte Symmetrie wird nun klar, denn da wir in obiger Rechnung die Rollen von X und Y vertauschen können, gilt auch die Formel

$$\begin{aligned}
I(X;Y) &= H(Y) - H(Y|X) \\
&= H(X) - \big(H(X,Y) - H(Y)\big) \\
&= H(X) + H(Y) - H(X,Y)\,,
\end{aligned}$$

aus der insbesondere folgt, daß $I(X;Y) = I(Y;X)$ ist.

Wenn wir bedingte Wahrscheinlichkeiten betrachten wollen, müssen wir auch hier wieder die Begriffe leicht abwandeln:

Definition: *a)* $p(x,y)$ und $q(x,y)$ seien zwei Wahrscheinlichkeitsverteilungen auf der Menge $A \times B$. Die *bedingte* KULLBACK-LEIBLER-*Distanz* zwischen p und q ist

$$D\big(p(y|x)\|q(y|x)\big) = \sum_{x\in A} p(x) \sum_{y\in B} p(y|x) \log_2 \frac{p(y|x)}{q(y|x)} \,.$$

b) Für drei Zufallsvariablen X, Y, Z ist die *bedingte wechselseitige Information von X und Y bei Kenntnis von Z*

$$I(X;Y|Z) = H(X|Z) - H(X|Y,Z) \,.$$

Für beide Größen gelten ähnliche Kettenregeln wie für die Entropie:

Lemma: *a)* $D\big(p(x,y)\|q(x,y)\big) = D\big(p(x)\|q(x)\big) + D\big(p(y|x)\|q(y|x)\big)$
b) Für $n+1$ Zufallsvariablen $X_1, \ldots, X_n$ und Y ist

$$I(X_1, \ldots, X_n; Y) = \sum_{i=1}^{n} I(X_i; Y|X_1, \ldots, X_{i-1}) \,.$$

Zum *Beweis* müssen wir in beiden Fällen einfach nachrechnen:

$$\begin{aligned}
a)\ D\big(p(x,y)\|q(x,y)\big) &= \sum_{x\in A}\sum_{y\in B} p(x,y) \log_2 \frac{p(x,y)}{q(x,y)} \\
&= \sum_{x\in A}\sum_{y\in B} p(x,y) \log_2 \frac{p(x)p(y|x)}{q(x)q(y|x)} \\
&= \sum_{x\in A}\sum_{y\in B} p(x,y) \log_2 \frac{p(x)}{q(x)} + \sum_{x\in A}\sum_{y\in B} p(x,y) \log_2 \frac{p(y|x)}{q(y|x)} \\
&= \sum_{x\in A} p(x) \log_2 \frac{p(x)}{q(x)} + \sum_{x\in A}\sum_{y\in B} p(x,y) \log_2 \frac{p(y|x)}{q(y|x)} \\
&= D\big(p(x)\|q(x)\big) + D\big(p(y|x)\|q(y|x)\big)
\end{aligned}$$

b) $I(X_1,\ldots,X_n;Y) = H(X_1,\ldots,X_n) - H(X_1,\ldots,X_n|Y)$

$$= \sum_{i=1}^{n} H(X_i|X_1,\ldots,X_{i-1}) - \sum_{i=1}^{n} H(X_i|X_1,\ldots,X_{i-1},Y)$$

$$= \sum_{i=1}^{n} I(X_i;Y|X_1,\ldots,X_{i-1})\,.$$

c) Berechnung der mittleren Entropie

In Abschnitt *a)* hatten wir die mittlere Entropie

$$H \underset{\text{def}}{=} \lim_{n\to\infty} \frac{H(X_1,\ldots,X_n)}{n}$$

eines stochastischen Prozesses $\mathcal{X} = (X_n)_{n\in\mathbb{N}}$ definiert; nach den Vorbereitungen aus Abschnitt *b)* können wir jetzt untersuchen, unter welchen Bedingungen sie existiert, und wie sie auch anders berechnet werden kann.

Dazu betrachten wir die Information, die uns die n-te Zufallsvariable X_n *neu* liefert, wenn wir ihre Vorgänger $X_1,\ldots,X_{n-1}$ bereits kennen, und lassen auch hier n gegen Unendlich gehen; falls der Grenzwert existiert, schreiben wir

$$H'(\mathcal{X}) \underset{\text{def}}{=} \lim_{n\to\infty} H(X_n|X_1,\ldots,X_{n-1})\,.$$

Satz: Für einen stationären stochastischen Prozess $\mathcal{X} = (X_n)_{n\in\mathbb{N}}$ existieren die Grenzwerte $H(\mathcal{X})$ und $H'(\mathcal{X})$ und sind gleich.

Der *Beweis* besteht aus drei Schritten:

1. Schritt: $H'(\mathcal{X})$ existiert

Da die Entropie durch Zusatzinformation höchstens kleiner werden kann, ist

$$H(X_{n+1}|X_1,X_2,\ldots,X_n) \le H(X_{n+1}|X_2,\ldots,X_n)\,.$$

Wegen der Stationarität des Prozesses können wir auf der rechten Seite alle Indizes um eins erniedrigen, ohne daß sich der Wert der bedingten Entropie ändert; daher ist auch

$$H(X_{n+1}|X_1,X_2,\ldots,X_n) \le H(X_n|X_1,X_2,\ldots,X_{n-1})$$

für alle $n \in \mathbb{N}$. Somit ist die Folge der $H(X_n|X_1,\ldots,X_{n-1})$ monoton fallend, und da auch bedingte Entropien nie negativ sind, ist sie nach unten beschränkt. Beides zusammen impliziert die Konvergenz.

2. Schritt: Konvergiert eine Folge $(x_n)_{n\in\mathbb{N}}$ von reellen Zahlen gegen einen Grenzwert a, so konvergiert auch die Folge der arithmetischen Mittel $y_n = \frac{1}{n}(x_1 + \cdots + x_n)$ gegen a (CESÀRO-Mittel).

Dazu müssen wir zeigen, daß es zu jedem $\varepsilon > 0$ ein $N \in \mathbb{N}$ gibt derart, daß $|a - y_n| < \varepsilon$ ist für alle $n \geq N$.

Da die Folge der x_n gegen a konvergiert, gibt es jedenfalls ein $M \in \mathbb{N}$, so daß $|a - x_n| < \frac{1}{2}\varepsilon$ ist für alle $n \geq M$. Für solche n ist dann

$$\begin{aligned}
|a - y_n| &= \left|\frac{1}{n}\sum_{i=1}^{n}(a - x_i)\right| \leq \frac{1}{n}\sum_{i=1}^{n}|a - x_i| \\
&= \frac{1}{n}\sum_{i=1}^{M-1}|a - x_i| + \frac{1}{n}\sum_{i=M}^{n}|a - x_i| \\
&< \frac{1}{n}\sum_{i=1}^{M-1}|a - x_i| + \frac{(n - M + 1)}{n}\frac{\varepsilon}{2} \leq \frac{1}{n}\sum_{i=1}^{M-1}|a - x_i| + \frac{\varepsilon}{2}\,.
\end{aligned}$$

Die Summe S über die ersten $M - 1$ Abweichungen hängt nicht von n ab; wir können daher leicht ein $M' \in \mathbb{N}$ finden, so daß $S/n < \varepsilon/2$ ist für alle $n \geq M'$. Ist n mindestens gleich dem Maximum N von M und M', so muß daher $|a - y_n| < \varepsilon$ sein, wie verlangt.

3. Schritt: Der Grenzwert $H(\mathcal{X})$ existiert und ist gleich $H'(\mathcal{X})$

Nach der Kettenregel für die Entropie ist

$$H(X_1,\ldots,X_n) = \sum_{i=1}^{n} H(X_i|X_1,\ldots,X_{i-1})\,;$$

dividieren wir beide Seiten durch n, sind wir genau in der Situation des zweiten Schritts, wobei links die Glieder jener Folge stehen, als deren Grenzwert $H(\mathcal{X})$ definiert ist, und rechts das arithmetische Mittel der ersten n Terme der Folge, deren Grenzwert nach dem ersten Schritt existiert und mit $H'(\mathcal{X})$ bezeichnet wurde.

Damit ist der Satz vollständig bewiesen. ∎

ERNESTO CESÀRO (1859–1906) wurde in Neapel geboren und wuchs auf in der nahe gelegenen Kleinstadt Torre Annunziata, wo sei Vater einen landwirtschaftlichen Betrieb mit Hofladen führte. Nach seiner Schulausbildung in Neapel studierte er ab 1873 in Liège Mathematik. Nach dem Tod seines Vaters kehrte er 1879 nach Torre Annunziata zurück um den Betrieb weiterzuführen. Dank eines Stipendiums konnte er ab 1882 sein Studium in Liège fortführen; teilweise studierte er auch in Paris und ab 1884 schließlich an der Universität Rom. Obwohl er bereits zahlreiche Arbeiten veröffentlicht hatte, wurde er dort erst 1887 promoviert und bekam dann gleich einen Lehrstuhl an der Universität von Palermo. 1891 folgte er einem Ruf an die Universität Neapel, wo er bis zu seinem Tod lehrte. Der Großteil seiner Arbeiten befaßt sich mit Differentialgeometrie; er leistete aber auch Beiträge zur Zahlentheorie, unter anderem etwa zur Primzahlverteilung.

Speziell für (zeitinvariante) MARKOV-Ketten läßt sich $H'(\mathcal{X})$ leicht berechnen: Wegen der MARKOV-Eigenschaft ist

$$H(X_n|X_1,\ldots,X_{n-1}) = H(X_n|X_{n-1})\,,$$

und wegen der Zeitinvarianz ist das für alle n gleich $H(X_2|X_1)$. Somit ist hier die Entropierate einfach die bedingte Entropie einer jeden Zufallsvariablen bei Kenntnis ihres Vorgängers.

§5: Anwendungen in der Kryptologie

Während des zweiten Weltkriegs beschäftigten sich die Bell Telephone Laboratories mit Systemen zur Geheimhaltung von Nachrichten. Auch SHANNON, der ab 1941 dort arbeitete, forschte auf diesem Gebiet und entwickelte parallel dazu die Informationstheorie. Beides veröffentlichte er erst nach dem Krieg im *Bell Systems Technical Journal;* die Informationstheorie 1948, die Kryptologie 1949.

Verschlüsselungsverfahren wurden schon in der Antike benutzt; in großem Umfang wurden sie später vom Militär, von Diplomaten und von Spionen verwendet. Heute sind die Sicherung der Kommunikation über das Internet sowie der Bankenbereich die wichtigsten Anwendungsgebiete.

a) Kryptosysteme und ihre Kryptanalyse

Bei jedem Verschlüsselungsverfahren, das in größerem Umfang genutzt wird, muß man davon ausgehen, daß es nicht lange geheim bleibt. Deshalb formulierte AUGUSTE KECKHOFFS in seiner 1883 im *Journal des sciences militaires* veröffentlichten zweiteiligen Arbeit *La cryptographie militaire* die heute nach ihm benannte Maxime, wonach die Sicherheit der Verschlüsselung nicht von der Geheimhaltung des *Verfahrens* abhängen darf, sondern nur von der eines (häufig zu wechselnden) *Schlüssels.*

Wir gehen der Einfachheit halber davon aus, daß der zu verschlüsselnde Text das gleiche Alphabet A benutzt wie der Klartext; das war zwar in der Geschichte häufig nicht der Fall, ist aber bei der Art von Anwendungen, mit denen wir es heute zu tun haben, meist sogar unvermeidlich. Die Elemente von A müssen dabei keine einzelnen Buchstaben sein; oft handelt es sich auch um Buchstabenblöcke oder, bei vielen heutigen Systemen, Blöcke von beispielsweise 128 Bit. Für jeden Schlüssel s aus einer (hinreichend großen) Schlüsselmenge S haben wir eine Verschlüsselungsvorschrift $T_s\colon A \to A$, von der wir annehmen, daß es sich um eine bijektive Abbildung handelt. Die Menge $\{T_s\colon A \to A | s \in S\}$ aller dieser Abbildungen ist unser Kryptosystem.

Da (nicht nur) wir nach KERCKHOFFS davon ausgehen müssen, daß dieses System bekannt wird, dürfen wir seine Sicherheit nur danach beurteilen, wie schwierig es für einen Gegner ist, die jeweils verwendete Abbildung T_s zu identifizieren. Dabei dürfen wir freilich nicht einfach untersuchen, wie schwierig es für *uns* wäre, diese Abbildung zu identifizieren: Ein ernstzunehmender Gegner wird häufig über Methoden verfügen, die nicht in der offenen Literatur dokumentiert sind, und ihm wird auch, gerade wenn es sich um eine Regierungsstelle handelt, oft Hardware zur Verfügung stehen, die auf dem Markt entweder nicht oder nur zu sehr hohen Preisen erhältlich ist.

SHANNON untersucht daher nicht, wieviel Information ein konkreter Gegner aus einem Kryptogramm ziehen kann, sondern er untersucht, wieviel Information darin enthalten ist – unabhängig davon, ob sie ein Gegner mit realistischem Aufwand nutzen kann.

Man kann das auch so formulieren, daß wir den Gegner deutlich überschätzen, indem wir annehmen, daß ihm unbegrenzte Resourcen zur Verfügung stehen; er kann also *jeden* endlichen Algorithmus ausführen, unabhängig von physikalischen und sonstigen Einschränkungen. Insbesondere könnte er beispielsweise eine Substitutionschiffre, die das Alphabet einer beliebigen Permutation unterwirft, einfach dadurch lösen, daß er alle $26! \approx 4 \cdot 10^{26}$ Möglichkeiten ausprobiert. Sofern dabei in genau einem Fall sinnvoller Klartext entsteht (was bei deutschen Texten mit ziemlicher Sicherheit der Fall ist), hat er die Lösung gefunden. (Die besten heute verfügbaren Supercomputer bräuchten dazu deutlich mehr als die Zeit, die unser Universum bislang existiert; ein erfahrener Kryptanalytiker freilich bräuchte, mit anderen Methoden, nur wenige Minuten, um diese Chiffre zu knacken.) Es ist klar, daß von einem realen Gegner keine größere Gefahr ausgehen kann, als von diesem idealisierten Gegner; ein Kryptosystem ist also sicher, wenn es gegen einen solchen Gegner sicher ist.

In der Kryptologie bezeichnet man diesen idealisierten Gegner als den BAYESschen Gegner; sein Name ist abgeleitet von dem englischen Theologen THOMAS BAYES, der als erster das Entscheidungsprinzip formulierte, wonach unter mehreren möglichen Hypothesen diejenige als wahr anzusehen sei, die vor dem Hintergrund seines vorhandenen Wissens die größte Wahrscheinlichkeit hat. Auf die Kryptanalyse angewandt bedeutet dies, daß man sich unter allen möglichen Schlüsseln für den entscheidet, der angesichts der Information, die der Chiffretext sowie das Vorwissen über die Quelle bieten die größte Wahrscheinlichkeit hat.

THOMAS BAYES (1702–1761) wurde in London geboren als ältestes von sieben Kindern eines der ersten nonkonformistischen Pastoren Englands. Da die englischen Universitäten Oxford und Cambridge keine Nonkonformisten akzeptieren, mußte er zum Studium 1719 nach Schottland an die Universität Edinburgh, wo er sich für Logik und Theologie immatrikulierte. Nach seinen späteren Äußerungen muß er sich auch bereits damals oder kurz danach mit Mathematik beschäftigt haben. Wie sein Vater wurde er Geistlicher; seine mathematischen Arbeiten, z.B. über die Grundlagen der Analysis,

erschienen zu seinen Lebzeiten nur anonym. Trotzdem wurde er 1742 fellow der Royal Society, die 1764 auch posthum seinen *Essay towards solving a problem in the doctrine of chances* veröffentlichte.

Wenn wir einen Klartext $a_1 a_2 \dots a_N \in A^N$ mit dem Schlüssel $s \in S$ verschlüsseln zu einem Kryptogramm $c_1 c_2 \dots c_N \in A^N$ mit $c_i = T_s(a_i)$, kann ein BAYESscher Gegner daher alle Schlüssel $s \in S$ durchprobieren; wie schon sein Name vermuten läßt, wird es sich dann für den Schlüssel entscheiden, für den der resultierende Klartext gemäß den BAYESschen Formeln die höchste Wahrscheinlichkeit hat. Diese Wahrscheinlichkeit berechnet er auf Grund seines Vorwissens über die Struktur der zu erwarteten Nachrichten, also beispielsweise anhand der statistischen Besonderheiten der Klartextsprache.

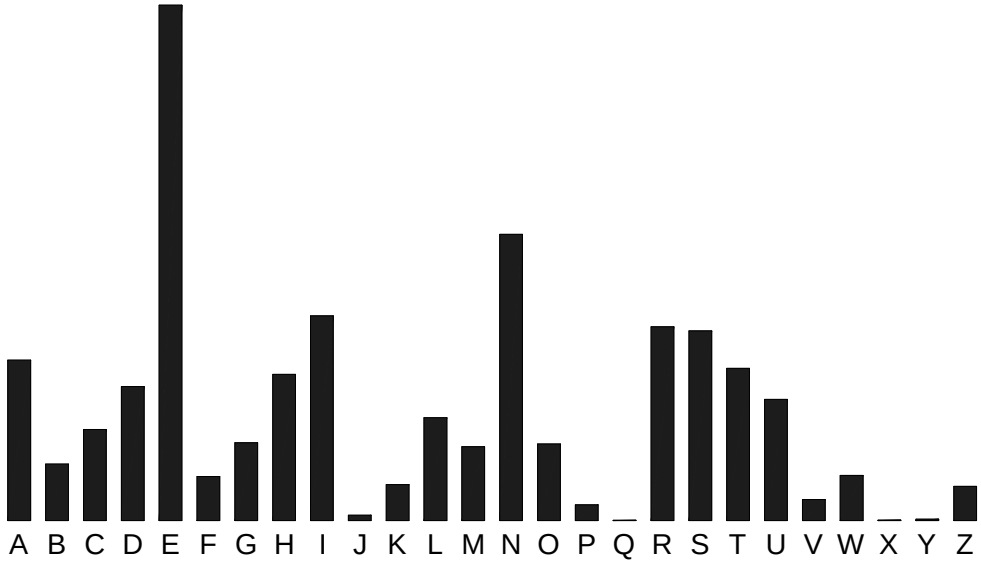
Da wir keine BAYESschen Gegner sind, können wir seine Arbeitsweise nur an sehr einfachen Beispielen nachvollziehen und müssen auch mit sehr einfachen Modellen für die Sprache arbeiten. Wir betrachten nur deutschen Klartext und dazu die beiden bereits bekannten statistischen Modelle: Zum einen modellieren wir die Sprache als eine Folge unabhängiger Zufallsvariablen, die uns jeweils einen Buchstaben liefern, zum anderen durch eine MARKOV-Kette erster Ordnung.

Die prozentualen Häufigkeiten der einzelnen Buchstaben in deutschen Klartext, ermittelt anhand der 260 238 Buchstaben aus JEAN PAULS Roman *Dr. Katzenbergers Badereise,* sind in der folgenden Tabelle dem weiter unten stehenden Balkendiagramm zu finden; für die Paarhäufigkeiten habe ich mich auf ein Diagramm beschränkt, dessen x- und y-Achse mit den Positionen der Buchstaben im Alphabet beschriftet sind.

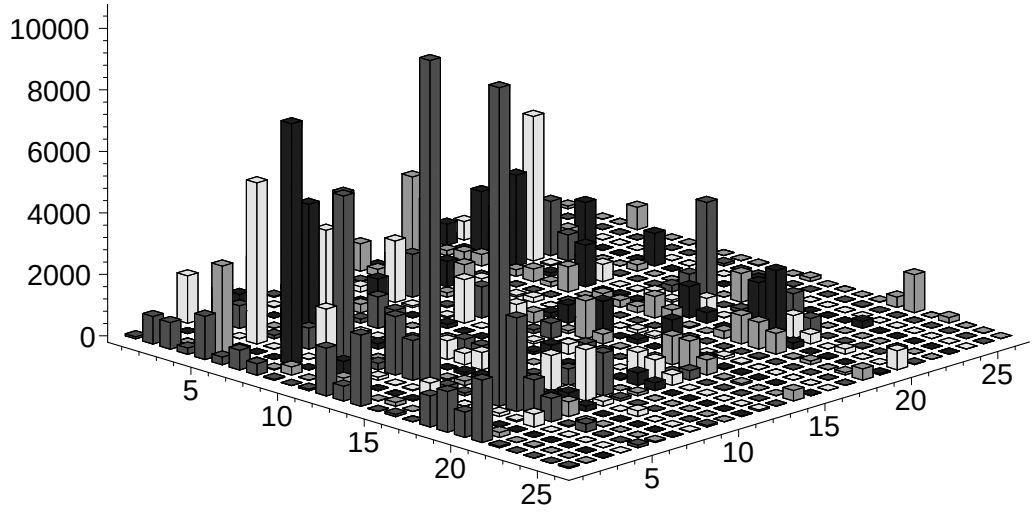
A	B	C	D	E	F	G	H	I
5,75	2,03	3,27	4,81	18,49	1,58	2,79	5,25	7,35

J	K	L	M	N	O	P	Q	R
0,19	1,29	3,69	2,65	10,27	2,75	0,57	0,01	6,95

S	T	U	V	W	X	Y	Z
6,81	5,46	4,35	0,76	1,62	0,02	0,05	1,23



Buchstabenhäufigkeiten in deutschem Text



Häufigkeiten von Buchstabenpaaren in deutschem Text

Der BAYESsche Gegner, der mit einem unserer beiden Modelle arbeitet, berechnet die Wahrscheinlichkeit eines Worts $a_1 a_2 \ldots a_N$ somit via Buchstabenhäufigkeiten als $p(a_1)p(a_2)\cdots p(a_N)$ und via MARKOV-Ketten als $p(a_1)p(a_2|a_1)\cdots p(a_N|a_{N-1})$.

b) Ein einfaches Beispiel

Um ihm bei der Arbeit zusehen zu können, müssen wir ein Kryptosystem mit wenigen Schlüsseln betrachten; wir nehmen die sogenannte CAESAR-Chiffre. CAESAR verschlüsselte seine geheimen Nachrichten, indem er die Buchstaben des Alphabets zyklisch um drei Positionen nach rechts verschob, also

$$A \to D, \quad B \to E, \ \ldots, \ W \to Z, \quad X \to A, \quad Y \to B, \quad Z \to C\,.$$

Heute bezeichnet man jede zyklische Verschiebung des Alphabets als CAESAR-Chiffre; mit unserem heutigen Alphabet haben wir daher 26 solche Chiffren, von denen allerdings eine die identische Abbildung ist.

Angenommen, ein BAYESscher Gegner empfängt das mit einer dieser Transformationen verschlüsselte Kryptogramm

QJULIILPPRUJHQJUDXHQ .

Dann wird er darauf die Inverse von jeder der 26 CAESAR-Verschiebungen anwenden und die Wahrscheinlichkeit der einzelnen Entschlüsselungen nach einer der beiden obigen Formeln berechnen. Um seine Arbeitsweise besser zu erkennen, wollen wir diese Wahrscheinlichkeiten nicht nur für die Entschlüsselung der gesamten Nachricht berechnen, sondern auch bestimmen, welcher Klartext anhand der ersten n Buchstaben am wahrscheinlichsten ist. Die erste Entschlüsselung in jeder Zeile benutzt nur Buchstabenhäufigkeiten, die zweite arbeitet mit dem MARKOV-Modell, und nach jeder Entschlüsselung steht jeweils deren relative Wahrscheinlichkeit für dieses konkrete Kryptogramm.

E	18,498	E	18,498
RE	25,379	ER	54,251
NAT	25,832	ERK	40,153
NATE	71,640	NATE	77,103
ANGRI	50,664	ANGRI	58,662
NATEV S	58,718	ANGRI F	95,260
NATEV SS	78,445	ANGRI FF	99,281
NATEV SSV	33,482	ANGRI FFI	99,311
ANGRI FFIM	42,358	ANGRI FFIM	99,974
ANGRI FFIMM	31,116	ANGRI FFIMM	99,997

FSLWN KKNRR T	35,311	ANGRI FFIMM O	100,000
ANGRI FFIMM OR	43,708	ANGRI FFIMM OR	100,000
ANGRI FFIMM ORG	33,840	ANGRI FFIMM ORG	100,000
ANGRI FFIMM ORGE	65,818	ANGRI FFIMM ORGE	100,000
ANGRI FFIMM ORGEN	84,217	ANGRI FFIMM ORGEN	100,000
ANGRI FFIMM ORGEN G	73,487	ANGRI FFIMM ORGEN G	100,000
ANGRI FFIMM ORGEN GR	51,676	ANGRI FFIMM ORGEN GR	100,000
NATEV SSVZZ BETRA TEN	62,256	ANGRI FFIMM ORGEN GRA	100,000
NATEV SSVZZ BETRA TENH	66,592	ANGRI FFIM MORGENG RAU	100,000
ANGRI FFIMM ORGEN GRAUE	57,125	ANGRI FFIMM ORGEN GRAUE	100,000
ANGRI FFIMM ORGEN GRAUE N	70,387	ANGRI FFIMM ORGEN GRAUE N	100,000

Mit Buchstabenhäufigkeiten allein schwankt der BAYESsche Gegner also noch ziemlich lange zwischen zwei Schlüsseln; beachtet er auch Kontakthäufigkeiten, hat er schon nach fünf Buchstaben die richtige Entschlüsselnung und ab dem sechsten Buchstaben weiß er das mit einer Wahrscheinlichkeit von über 95%.

Als nächstes Kryptogramm betrachten wir PWDUYFSFQDXJ. Hier ergeben sich folgende wahrscheinlichste Entschlüsselungen:

E	18,498	E	18,498
EL	20,611	EL	27,369
ELS	36,252	ELS	61,022
ZGNE	20,524	MTAR	27,712
ZGNEI	51,027	MTARV	32,544
NUBSW D	25,059	DKRIM T	46,246
ELSJN UH	42,422	KRYPT AN	63,235
ELSJN UHU	44,039	KRYPT ANA	72,856
DKRIM TGTE	81,297	KRYPT ANAL	53,404
DKRIM TGTER	85,742	DKRIM TGTER	99,705
DKRIM TGTER L	87,118	DKRIM TGTER L	98,089
ZGNEI PCPAN HT	75,437	KRYPT ANALY SE	99,797

Hier gelingt die Entschlüsselung also nur mittels Kontakthäufigkeiten, und auch da nur dank des guten Doktors STRYKIUS, der dafür sorgt, daß das Digramm „RY“ in *Dr. Katzenbergers Badereise* mit positiver Wahrscheinlichkeit auftritt: Bei vielen anderen Referenztexten tritt diese

Buchstabenkombination nie auf, so daß die Wahrscheinlichkeit eines jeden Worts, das sie enthält auf null gesetzt würde.

Man muß freilich beachten, daß der ideale BAYESsche Gegner nicht mit der Auszählung eines einzelnen Texts arbeitet, sondern *die* korrekte Wahrscheinlichkeitsverteilung kennt – wie immer diese auch definiert sein mag.

Es ist kein Zufall, daß der BAYESsche Gegner einbuchstabige Nachrichten immer als „E“ entschlüsselt: Für einbuchstabige Texte gibt es nach beiden Wahrscheinlichkeitsmodellen keine bessere Alternative. Entsprechend einfach werden auch Wörter erkannt, die mit „E“ beginnen und viele häufige Buchstaben und Buchstabenkombinationen enthalten wie beispielsweise im Kryptogramm JSIQNHMPJNY, wo beide Modelle sofort den richtigen Schlüssel finden:

E	18,498	E	18,498
EN	43,447	EN	66,315
END	49,997	END	96,907
ENDL	60,346	ENDL	93,935
ENDLI	89,043	ENDLI	99,306
ENDLI C	89,309	ENDLI C	99,995
ENDLI CH	96,787	ENDLI CH	100,000
ENDLI CHK	92,988	ENDLI CHK	100,000
ENDLI CHKE	98,564	ENDLI CHKE	100,000
ENDLI CHKEI	99,529	ENDLI CHKEI	100,000
ENDLI CHKEI T	99,939	ENDLI CHKEI T	100,000

Auch nicht zu exotische Substantive mit Artikel sind meist problemlos: IJWYNXHM führt zu

E	18,498	E	18,498
DE	22,057	DE	38,733
DER	38,490	DER	87,654
DERT	51,481	DERT	96,107
DERTI	75,471	DERTI	99,393
DERTI S	88,877	DERTI S	99,717
DERTI SC	87,934	DERTI SC	99,999
DERTI SCH	90,031	DERTI SCH	100,000

und PUQEFGQTXQ zu

E	18,498	E	18,498
IN	23,056	CH	40,135
DIE	41,643	DIE	43,465
DIES	64,018	DIES	70,969
DIEST	69,681	DIEST	90,466
DIEST U	53,199	DIEST U	76,378
DIEST UE	83,884	DIEST UE	91,854
DIEST UEH	87,958	DIEST UEH	97,642
DIEST UEHL	96,798	DIEST UEHL	99,523
DIEST UEHLE	99,372	DIEST UEHLE	99,978

Nur der sächliche Artikel ist etwas problematischer: EBTIBVT wird entschlüsselt als

E	18,498	E	18,498
HE	29,769	HE	28,458
EBT	20,051	DAS	47,768
EBTI	34,481	DASH	32,720
HEWLE	45,218	DASHA	75,568
DASHA U	47,297	DASHA U	98,212
DASHA US	57,475	DASHA US	99,609

Hier gibt es also Schwierigkeiten mit den Buchstabenhäufigkeiten allein, da der Text aus eher nicht so häufigen Buchstaben zusammengesetzt ist. Dieses Problem tritt auch bei anderen Texten auf wie etwa HAQQNAA mit

E	18,498	E	18,498
LE	20,611	UN	25,577
LEU	29,391	UND	69,721
LEUU	24,785	UNDD	88,852
LEUUR	44,254	UNDDA	97,663
LEUUR E	68,174	UNDDA N	99,279
LEUUR EE	81,555	UNDDA NN	99,890

und selbst bei einem so zentralen kurpfälzerischen Wort wie FNHZNTRA, wo wir auf die folgenden Entschlüsselungen kommen:

E	18,498	E	18,498
EM	15,918	EM	21,504
LTN	19,493	CKE	37,812
AICU	20,640	SAUM	70,182
AICUI	28,589	SAUMA	97,483
AICUI O	29,941	SAUMA G	98,658
SAUMA GE	52,561	SAUMA GE	99,995
SAUMA GEN	78,116	SAUMA GEN	100,000

Zusammenfassend können wir festhalten, daß dem BAYESschen Gegner zumindest mit Kontakthäufigkeiten bereits wenige Buchstaben zur korrekten Entschlüsselung reichen. Beachtet man noch, daß sich weder ein BAYESsche Gegner noch ein realer Kryptanalytiker auf Kontakthäufigkeiten beschränken muß, sondern auch Sprachkenntnisse einsetzt, die sich auf Trigramme, Worte, Wortpaare und so weiter beziehen, kommen sie mit noch weniger Buchstaben aus, als obige Beispiele nahelegen.

Als Kuriositär am Rande sei noch erwähnt, daß der BAYESsche Gegner auch ohne Chiffretext bereits wahrscheinlichste „Entschlüsselungen“ findet: Falls er nur mit Buchstabenhäufigkeiten arbeitet, ist das natürlich eine Folge von lauter „E“s, schon bei Kontakthäufigkeiten ergeben sich aber interessantetere Ergebnisse, die hier bis zur Nachrichtenlänge acht aufgeführt sind:

n	Text	Wahrscheinlichkeit
1	e	0.184945672750
2	en	0,042051665485
3	ich	0,011662006378
4	ende	0,003537155688
5	eiche	0,000825895113
6	endich	0,000201549003
7	endende	0,000067649431
8	eichende	0,000015795555

c) Allgemeine Vorgehensweise des Bayesschen Gegners

Nach diesen Beispielen wollen wir uns allgemein überlegen, wie der BAYESsche Gegner vorgeht.

Als erstes braucht er Wissen oder zumindest Annahmen über den Inhalt der Nachrichten: In den obigen Beispielen handelte es sich um deutschen Klartext, verschlüsselt auf der Grundlage von 26 Buchstaben ohne Zwischenräume und Satzzeichen. Das ist natürlich nicht mehr der typische Fall für heutige Kryptanalyse. Dort geht es eher um Dokumente von Textverarbeitungs-, Tabellenkalkulations- oder Datenbankprogrammen oder um ausführbare Programme für ein gegebenes Betriebssystem oder ähnliches.

In jedem dieser Fälle verschafft er sich als erstes durch Auszählen eine Wahrscheinlichkeitsverteilung für die möglichen Klar„texte“ der Länge N und ordnet dadurch jedem solchen Klartext W eine

$$\textit{Klartextwahrscheinlichkeit } p_K(W)$$

zu. Diese kann beispielsweise wie oben über Buchstabenhäufigkeiten oder über Kontakthäufigkeiten definiert sein, aber auch kompliziertere Ansätze sind möglich.

Außerdem ordnet er jedem Schlüssel s aus dem Schlüsselraum S eine

$$\textit{Schlüsselwahrscheinlichkeit } p_S(s)$$

zu. Bei einem gut gemanagten Kryptosystem sollten alle Schlüssel mit gleicher Wahrscheinlichkeit auftreten, so daß

$$p_S(s) = \frac{1}{\#S} \quad \text{für alle } s \in S$$

ist, aber in der Praxis kann der Gegner oft große Vorteile aus der Tatsache ziehen, daß dies nicht der Fall ist. (Wie zufällig sind Ihre Paßwörter?)

Falls nun ein Chiffretext $C \in A^N$ aufgefangen wird, muß der BAYESsche Gegner berechnen, wie wahrscheinlich jeder Schlüssel $s \in S$ vor dem Hintergrund dieses Chiffretexts ist. Dazu müssen bedingte Wahrscheinlichkeiten berechnet werden, also muß man zunächst die Wahrscheinlichkeit des Chiffretexts $C \in A^N$ kennen.

C entsteht aus einem Klartext $W \in A^N$ durch Verschlüsselung mit einem Schlüssel $s \in S$; da der Schlüssel vom Klartext unabhängig sein sollte, ist die Wahrscheinlichkeit für das Zusammentreffen von Klartext W und Schlüssel s gleich

$$p_K(W) \cdot p_S(s)\,.$$

Die Wahrscheinlichkeit, einen bestimmten Chiffretext C zu empfangen, ist also *a priori* gleich

$$p_{Ch}(C) = \sum_{\substack{(W,s) \in A^N \times S \\ T_s(W) = C}} p_K(W) \cdot p_S(s) \,.$$

Die Wahrscheinlichkeit, daß C auftritt und mit Schlüssel $s \in S$ verschlüsselt wurde, ist entsprechend gleich

$$p_{Ch,S}(C,s) = \sum_{\substack{W \in A^N \\ T_s(W) = C}} p_K(W) \cdot p_S(s) \,,$$

wobei die Summe hier im allgemeinen wegen der Injektivität von T_s nur einen Summanden hat. (T_s muß nicht injektiv auf A^N sein, sondern nur auf der Teilmenge der tatsächlich vorkommenden Nachrichten.)

Die eigentlich interessante Wahrscheinlichkeit, die Wahrscheinlichkeit, daß ein gegebener Chiffretext C durch Verschlüsselung mit $s \in S$ entstanden ist, berechnet sich nun als

$$p_{S|Ch}(s|C) = \frac{p_{Ch,S}(C,s)}{p_{Ch}(C)} \,.$$

Der BAYESsche Gegner entscheidet sich bei seinem Entschlüsselungsversuch bekanntlich für einen Schlüssel, der diese bedingte Wahrscheinlichkeit maximiert. Um einen solchen Schlüssel zu bestimmen, muß er $p_{S|Ch}(s|C)$ möglicherweise für alle Schlüssel $s \in S$ berechnen, jedoch ist dies angesichts seiner unbegrenzten Rechenfähigkeit kein Problem.

Solange es sich nur um eine Nachricht dreht, wird der BAYESsche Gegner nicht in erster Linie am Schlüssel interessiert sein, sondern am wahrscheinlichsten Klartext. Hier ist entsprechend die Wahrscheinlichkeit für das Zusammentreffen von Klartext W und Chiffretext C gleich

$$p_{Ch,K}(C,W) = \sum_{\substack{s \in S \\ T_s(W) = C}} p_K(W) \cdot p_S(s) \,,$$

und die bedingte Wahrscheinlichkeit für Klartext W, nachdem Chiffretext C aufgefangen wurde, ist

$$p_{K|Ch}(W|C) = \frac{p_{Ch,K}(C,W)}{p_{Ch}(C)} \,.$$

Definition: Eine BAYESsche Entscheidungsfunktion ist eine Familie von Abbildungen

$$\delta_N \colon A^N \to A^N$$

mit der Eigenschaft, daß für jeden Chiffretext $C \in A^N$ gilt:

$$p_{K|Ch}(\delta_N(C)|C) = \max_{W \in A^N} p_{K|Ch}(W|C)\,.$$

Weniger formal ausgedrückt: Für jeden Chiffretext $C \in A^N$ ist $\delta_N(C)$ ein Klartext mit höchstmöglicher Wahrscheinlichkeit – idealerweise *der* Klartext mit höchstmöglicher Wahrscheinlichkeit, allerdings könnte es auch mehrere Klartexte mit gleicher Wahrscheinlichkeit geben, so daß $\delta_N(C)$ im allgemeinen durch die obige Definition nicht eindeutig bestimmt ist. Trotzdem ist klar, daß man nichts besseres tun kann, als mit einer solchen BAYESschen Entscheidungsfunktion zu arbeiten; wenn dies zu nichts führt, ist ein Kryptosystem sicher.

d) Perfekte Sicherheit

Am allersichersten ist ein Kryptosystem, wenn der BAYESsche Gegner aus dem aufgefangenen Chiffretext C *überhaupt keine* Information gewinnen kann, die über seine ursprünglich gewählte Wahrscheinlichkeitsfunktion p_K hinausgeht, wenn also die bedingten Wahrscheinlichkeiten, die er in Kenntnis von C errechnet, gleich den ursprünglichen Wahrscheinlichkeiten sind:

Definition: Ein Kryptosystem $\{T_s \mid s \in S\}$ über dem Alphabet A hat *perfekte Sicherheit* für Nachrichten der Länge N, wenn für jeden Chiffretext $C \in A^N$ und jeden Klartext $W \in A^N$ gilt:

$$p_{K|Ch}(W|C) = p_K(W)\,.$$

Satz: Falls ein Kryptosystem $\{T_s \mid s \in S\}$ perfekte Sicherheit für Nachrichten der Länge N hat, ist die Anzahl der Schlüssel $s \in S$ mindestens gleich der Anzahl der möglichen (d.h. mit Wahrscheinlichkeit $p_K(W) > 0$ auftretenden) Klartexte der Länge N.

Beweis: K^+ sei die Menge aller Klartexte aus A^N, die mit positiver Wahrscheinlichkeit auftreten, und C^+ sei die Menge der mit positiver

Wahrscheinlichkeit auftretenden Chiffretexte. Für jedes feste $W \in K^+$ und jedes $C \in C^+$ muß es dann einen Schlüsses $s \in S$ geben, so daß $T_s(W) = C$ ist; denn sonst wäre $p_{K|Ch}(W|C) = 0$, aber $p_K(W) > 0$; der Gegner könnte also aus dem Auftreten von C Information gewinnen. Damit muß es mindestens so viele Schlüssel geben, wie es Chiffretexte in C^+ gibt. Für jeden Schlüssel $s \in S$ ist aber T_s eine injektive Abbildung von K^+ nach C^+, d.h. es muß erst recht so viele Schlüssel geben, wie es mögliche Klartexte gibt.

■

Ein absolut sicheres Verfahren mit entsprechend großer Schlüssellänge läßt sich leicht auf Grund der CAESAR-Chiffre konstruieren: Wir nehmen als Schlüssel eine Zufallsfolge von Buchstaben, deren Länge größer ist als die zu erwartende Länge der zu übermittelten Nachrichten. In der Praxis geschah dies früher dadurch, daß die Seiten eines Schreibblocks mit Zufallsfolgen von Buchstaben bedruckt wurden, wobei jeder Block in Auflage zwei hergestellt wurde: je ein Exemplar für Sender und Empfänger. Der Sender nimmt für seine erste Nachricht die erste Seite des Blocks und „addiert“ die Buchstabenfolge nach Art einer CAESAR-Chiffre zu den Buchstaben seiner Nachricht – wobei nun freilich jeder Buchstabe mit einer neuen CAESAR-Chiffre verschlüsselt wird, zum Beispiel nach dem Schema

```
  ANGRI FFIMM ORGEN GRAUE N
+ KCHQR OFVFN FVSLA XRQBV E
= LQOIA ULESA UNZQO EJRWA S
```

Nach Verschlüsselung der Nachricht wird das erste Blatt des Blocks vernichtet, nach Entschlüsselung der Nachricht auch beim Empfänger. Man bezeichnet das Verfahren daher als *one time pad.*

Da wir nicht wissen, wie ein Gegner vorgeht, können wir nicht ausschließen, daß er irgendwie auf den korrekten Schlüssel stößt und damit die Nachricht entschlüsselt. Das nützt ihm aber nichts, denn es gibt noch viele andere Klartexte, die zu diesem Kryptogramm passen: Wenn er andere Schlüssel ausprobiert, erhält er beispielsweise auch die

Entschlüsselung

```
  LQOIA ULESA UNZQO EJRWA S
− JYFUT PJSXN PZTVJ MWWCG J
= BRING EBLUM ENFUE RMUTT I
```

und entsprechend läßt sich auch jeder andere Klartext mit 21 Buchstaben produzieren; er kann aus dem Kryptogramm also keine weitere Information ziehen, als daß der Klartext 21 Buchstaben lang war.

e) Die Mehrdeutigkeit eines Schlüssels

In der Praxis ist es nicht unbedingt notwendig, daß der Gegner aus einem aufgefangenen Chiffretext *überhaupt keine* Information ziehen kann; es reicht, wenn er *nicht genügend* Information bekommt.

Als nächstes wollen wir daher abschätzen, wieviel Information der (besonders gefährliche) BAYESsche Gegner aus einem Chiffretext ziehen kann. Besonders wichtig ist der Schutz des Schlüssels s, denn sobald dieser bekannt ist, können bis zum nächsten Schlüsselwechsel alle Nachrichten entschlüsselt werden.

Die Information, die der Gegner zur Rekonstruktion des Schlüssels gewinnen muß, läßt sich nach den Vorarbeiten des letzten Paragraphen leicht quantifizieren: Es ist die Entropie

$$H_S = -\sum_{s\in S} p_S(s)\log_2 p_S(s)\,.$$

Das Lemma über die Maximalität der Entropie bei Gleichverteilung zeigt also noch einmal die eigentlich auch so selbstverständliche Tatsache, daß ein Kryptosystem umso sicherer ist, je mehr die Wahrscheinlichkeitsverteilung der Schlüssel einer Gleichverteilung entspricht.

Der BAYESsche Gegner kennt die Entropie

$$H_{Ch} = -\sum_{C\,\in\,A^N} p_{Ch}(C)\log_2 p_{Ch}(C)$$

der Menge aller Chiffretexte der Länge N; nachdem er eine Nachricht C aufgefangen hat, hat er im Mittel diese Information gewonnen.

Um daraus Informationen über den Schlüssel zu erhalten, berechnet er zunächst die Entropie der Menge aller Paare (C, s):

$$\begin{aligned}
H_{Ch,S} &= -\sum_{(C,s)\in A^N\times S} p_{Ch,S}(C,s)\log_2 p_{Ch,S}(C,s)\\
&= -\sum_{(C,s)\in A^N\times S} p_{Ch,S}(C,s)\big(\log_2 p_{Ch}(C)+\log_2 p_{S|Ch}(s|C)\big)\\
&= -\sum_{(C,s)\in A^N\times S} p_{Ch,S}(C,s)\log_2 p_{Ch}(C)\\
&\quad -\sum_{(C,s)\in A^N\times S} p_{Ch,S}(C,s)\log_2 p_{S|Ch}(s|C)\\
&= -\sum_{C\in A^N} p_{Ch}(C)\log_2 p_{Ch}(C)\\
&\quad -\sum_{(C,s)\in A^N\times S} p_{Ch,S}(C,s)\log_2 p_{S|Ch}(s|C)\\
&= H_{Ch} - \sum_{(C,s)\in A^N\times S} p_{Ch,S}(C,s)\log_2 p_{S|Ch}(s|C)\,.
\end{aligned}$$

Die letzte Summe in der letzten Zeile bezeichnen wir als *bedingte Entropie* oder *Mehrdeutigkeit*

$$H_{S|Ch} \underset{\text{def}}{=} -\sum_{(C,s)\in A^N\times S} p_{Ch,S}(C,s)\log_2 p_{S|Ch}(s|C)$$

des Schlüssels bei vorliegendem Chiffretext; diese Information muß sich der Gegner verschaffen, um den Schlüssel zu bestimmen. Sobald $H_{S|Ch} = 0$ ist, kennt er den Schlüssel, denn da in der obigen Summe alle Summanden kleiner oder gleich null sind, kann $H_{S|Ch}$ nur dann gleich null sein, wenn jeder einzelne Summand verschwindet, wenn also für jeden Schlüssel $s \in S$ entweder $p(C, s) = 0$ ist, womit der Schlüssel mit Sicherheit ausgeschlossen ist, oder $\log_2 p(s|C) = 0$, d.h. $p(s|C) = 1$, womit der Schlüssel s mit Sicherheit richtig ist.

Um die Information abzuschätzen, die der Gegner maximal aus einem Chiffretext gewinnen kann, müssen wir also $H_{S|Ch}$ berechnen. Für ein beliebiges Kryptosystem und beliebige Wahrscheinlichkeitsverteilung

der Klartexte können wir offensichtlich nichts konkreteres hinschreiben als die definierende Summe. Die für die gesamte Informationstheorie grundlegende Idee von CLAUDE SHANNON war es, spezifischere Aussagen nicht über ein spezielles, sondern über das *durchschnittliche* Kryptosystem zu machen; diese Aussagen sollten dann *im wesentlichen* für die meisten Kryptosysteme gelten.

f) Randomisierung

Um von Durchschnitten reden zu können, müssen wir zunächst einige Annahmen machen über das Verhalten der Klartexte. Wir betrachten wieder Klartexte über einem Alphabet A aus n Buchstaben; für $x \in A$ sei $p(x)$ die Wahrscheinlichkeit, mit der der Buchstabe x auftritt, und für $(x, y) \in A^2$ sei $p(x|y)$ die bedingte Wahrscheinlichkeit dafür, daß er auf ein y folgt.

Wir hatten bislang immer angenommen, daß etwa die Buchstabenhäufigkeiten, die wir durch Auszählen eines hinreichend langen Texts erhalten, in hinreichend guter Näherung mit denen übereinstimmen, die wir in einem gegebenen Klartext vorfinden. Dies müssen wir für die folgenden Überlegungen etwas präziser fassen:

Definition: Für einen gegebenen Klartext $W = w_1 \dots w_N \in A^N$ sei

$$m_x(W) = \#\big\{ i \le N \;\big|\; w_i = x \big\}$$

und

$$m_{xy}(W) = \#\big\{ i \le N - 1 \;\big|\; w_i = x \text{ und } w_{i+1} = y \big\} .$$

$m_x(W)$ und $m_{xy}(W)$ zählen also, wie oft der Buchstabe x *bzw.* das Buchstabenpaar xy in W vorkommen. Idealerweise sollte für einen hinreichend langen Text W und zwei Buchstaben $x, y \in A$ gelten

$$p(x) \approx \frac{m_x(W)}{N} \quad \text{und} \quad p(x|y) \approx \frac{m_{yx}(W)}{m_y(W)} .$$

Exakt kann das natürlich schon aus zahlentheoretischen Gründen nicht gelten, aber wir wollen verlangen, daß die Abweichung für große N mit sehr kleiner Wahrscheinlichkeit größer als eine vorgebbare Schranke δ ist:

Definition: Eine Quelle für Klartexte heißt *ergodisch,* wenn es zu zwei beliebig vorgebbaren reellen Zahlen $\delta, \varepsilon > 0$ stets eine natürliche Zahl N_0 gibt, so daß für alle $N \geq N_0$ und alle $x \in A$ für Texte $W \in A^N$ gilt

$$p\left(\left|\frac{m_x(W)}{N} - p(x)\right| > \delta\right) < \varepsilon\,.$$

Unsere Ansätze zur Kryptanalyse klassischer Systeme beruhten somit stets darauf, daß wir annahmen, die deutsche Sprache sei eine ergodische Quelle für Klartexte. Der Erfolg bei unseren Entschlüsselungsversuchen spricht dafür, daß diese Annahme nicht garzu falsch sein sollte. In der Tat zeigt man in der Stochastik, daß jede Quelle, bei der es zu jedem Buchstabenpaar (x, y) einen Text gibt, in dem y irgendwo hinter x steht, ergodisch ist. Da wohl niemand bezweifelt, daß es solche Texte zu jedem Buchstabenpaar gibt (für (q, y) etwa **q**uer durch Z**y**pern), kann so die Ergodizität auch bewiesen werden.

Wir wollen im folgenden Aussagen machen über die *durchschnittliche* ergodische Quelle. Es ist klar, daß wir keine Chance haben, alle ergodischen Quellen zu bestimmen und dann einen Mittelwert zu bilden; der folgende Satz von SHANNON zeigt aber, daß wir durch Zusammenfassen der Buchstaben zu hinreichend langen Wörtern erreichen können, daß die Wahrscheinlichkeitsverteilung sehr einfach ist: Die Wörter zerfallen in zwei Klassen S und O, so daß Wörter aus S (wie *selten*) praktisch nie vorkommen, während die restlichen Wörter (aus O wie *oft*) alle praktisch dieselbe Wahrscheinlichkeit haben. Aus technischen Gründen arbeiten wir nicht mit Wahrscheinlichkeiten, sondern mit der Entropie: Die buchstabenweise Entropie der Quelle ist, wenn wir wie in §1, *3)* von Kontakthäufigkeiten ausgehen, gleich

$$H = -\sum_{x\in A}\sum_{y\in A} p(x)p(y|x)\log_2 p(y|x)\,,$$

und wir fordern, daß sich diese Entropie gleichmäßig auf die $W \in O$ verteilen soll:

Satz: Für eine ergodische Quelle gibt es zu zwei beliebig vorgebbaren reellen Zahlen $\varepsilon, \eta > 0$ stets ein $N_0 \in \mathbb{N}$, so daß A^N für $N \geq N_0$ als

Vereinigung zweier Teilmengen O und S geschrieben werden kann mit den Eigenschaften

1.) $p(W \in S) < \varepsilon$

2.) $\left| \frac{-\log_2 p(W)}{N} - H \right| < \eta \quad$ für alle $W \in O$.

Zum *Beweis* wählen wir zunächst willkürlich eine natürliche Zahl N und eine reelle Zahl $\delta > 0$ und setzen dann

$$O \underset{\text{def}}{=} \left\{ W \in A^N \;\middle|\; \begin{array}{l} p(W) > 0 \text{ und für alle } x, y \in A \text{ ist} \\ \left| m_{xy}(W) - N p(x) p(y|x) \right| < N\delta \end{array} \right\}$$

und $S = A^N \setminus O$. Für $W \in O$ können wir dann $m_{xy}(W)$ schreiben als

$$m_{xy}(W) = N p(x) p(y|x) + N\delta_{xy} \quad \text{mit} \quad \left|\delta_{xy}\right| < \delta$$

und erhalten damit für $W = w_1 \ldots w_N$

$$\begin{aligned} p(W) &= p(w_1) \prod_{i=2}^{N} p(w_i|w_{i-1}) \\ &= p(w_1) \prod_{x \in A} \prod_{y \in A} p(y|x)^{m_{xy}(W)} \\ &= p(w_1) \prod_{x \in A} \prod_{y \in A} p(y|x)^{N p(x) p(y|x) + N\delta_{xy}} \; . \end{aligned}$$

Logarithmieren führt auf

$$\begin{aligned} -\log_2 p(w) &= -\log_2 p(w_1) - N \sum_{x \in A} \sum_{y \in A} p(x) p(y|x) \log_2 p(y|x) \\ &\qquad - N \sum_{x \in A} \sum_{y \in A} \delta_{xy} \log_2 p(y|x) \\ &= -\log p(w_1) + NH - N \sum_{x \in A} \sum_{y \in A} \delta_{xy} \log_2 p(y|x) \end{aligned}$$

und somit zur Ungleichung

$$\begin{aligned} \left| \frac{-\log_2 p(W)}{N} - H \right| &= \left| \frac{-\log_2 p(W)}{N} - \sum_{x \in A} \sum_{y \in A} \delta_{xy} \log_2 p(y|x) \right| \\ &< \frac{-\log_2 p(W)}{N} + \delta \sum_{x \in A} \sum_{y \in A} -\log_2 p(y|x) \; . \end{aligned}$$

Wählt man N hinreichend groß und δ hinreichend klein, so läßt sich dieser Ausdruck offensichtlich kleiner als jedes vorgegebene $\eta > 0$ machen, so daß die Eigenschaft 2.) erfüllt ist.

Für die Eigenschaft 1.) müssen wir nachrechnen, mit welcher Wahrscheinlichkeit eine Nachricht $W \in A^N$ in S liegt. Sie liegt genau dann in S, wenn sie nicht in O liegt, für mindestens ein Buchstabenpaar (x, y) muß also

$$\left|m_{xy}(W) - Np(x)p(y|x)\right| \geq N\delta$$

sein. Die Wahrscheinlichkeit hierfür ist sicherlich nicht größer als die Summe über alle Paare (x, y) für die entsprechenden Wahrscheinlichkeiten, d.h.

$$p(W \in S) \leq \sum_{x\in A}\sum_{y\in A} p\left(\left|m_{xy}(W) - Np(x)p(y|x)\right| \geq N\delta\right) .$$

Wegen der Ergodizität der Quelle können wir für jedes $\widetilde{\varepsilon} > 0$ ein $N_1 \in \mathbb{N}$ finden, so daß für $N \geq N_1$ gilt

$$p\left(\left|\frac{m_x(W)}{N} - p(x)\right| > \frac{\delta}{2}\right) < \widetilde{\varepsilon}$$

oder, was dasselbe ist,

$$p\big(|m_x(W) - Np(x)| > \frac{\delta}{2}N\big) < \widetilde{\varepsilon}\,.$$

Für große N wird, wieder wegen der Ergodizität der Quelle, auch die Häufigkeit $m_x(W)$ groß; falls wir N so groß wählen, daß wir die Wahrscheinlichkeit des Ereignisses $m_x(W) < N_1$ vernachlässigen können, ist also auch

$$p\left(\left|\frac{m_{xy}(W)}{m_x(W)} - p(y|x)\right| > \frac{\delta}{2}\right) < \widetilde{\varepsilon}\,.$$

Die komplementäre Wahrscheinlichkeit dafür, daß

$$|m_x(W) - Np(x)| < \frac{\delta}{2}N$$

bzw.

$$\left|m_{xy}(W) - m_x(W)p(y|x)\right| < \frac{\delta}{2}m_x(W) \leq \frac{\delta}{2}N$$

ist, beträgt jeweils mindestens $1 - \widetilde{\varepsilon}$; die Wahrscheinlichkeit dafür, daß beides eintritt, ist also mindestens

$$(1 - \widetilde{\varepsilon})^2 > 1 - 2\widetilde{\varepsilon}\,.$$

Falls die erste der beiden Ungleichungen erfüllt ist, gilt erst recht

$$|m_x(W)p(y|x) - Np(x)p(y|x)| < \frac{\delta}{2}Np(y|x) \leq \frac{\delta}{2}N$$

und damit nach der Dreiecksungleichung zusammen mit der zweiten Ungleichung

$$\big|m_{xy}(W) - Np(x)p(y|x)\big| < \frac{\delta}{2}N + \frac{\delta}{2}N = N\delta\,.$$

Diese Ungleichung ist somit mindestens mit Wahrscheinlichkeit $1 - 2\widetilde{\varepsilon}$ erfüllt; die komplementäre Wahrscheinlichkeit ist

$$p\big(\big|m_{xy}(W) - Np(x)p(y|x)\big| > N\delta\big) < 2\widetilde{\varepsilon}\,.$$

Dies können wir in die oben abgeleitete Formel

$$p(W \in S) \leq \sum_{x\in A}\sum_{y\in A} p\left(\big|m_{xy}(W) - Np(x)p(y|x)\big| \geq N\delta\right)$$

einsetzen und erhalten, da es n^2 Buchstabenpaare gibt,

$$p(W \in S) < 2n^2\widetilde{\varepsilon}\,.$$

Da n eine Konstante ist, müssen wir nun nur $\widetilde{\varepsilon}$ hinreichend klein wählen und erhalten dann, daß diese Wahrscheinlichkeit höchstens gleich ε ist, wie behauptet.

■

Die Mengen O und S aus dem gerade bewiesenen Satz lassen sich für eine gegebene Quelle natürlich nur schwer konstruieren, denn im allgemeinen muß N doch schon sehr groß gewählt werden, damit dieser Satz gilt. Wir können allerdings trotzdem Aussagen über die Größe von O und S machen: Für kleine Werte von ε und η machen wir keinen großen Fehler, wenn wir davon ausgehen, daß alle Nachrichten aus O mit *exakt* derselben Wahrscheinlichkeit vorkommen und daß Nachrichten aus S *nie* vorkommen. Dann ist für $W \in A^N$ also

$$p(W) = \begin{cases} 1/\#O & \text{falls } W \in O \\ 0 & \text{falls } W \in S \end{cases}$$

und die Entropie der Quelle ist

$$-\sum_{W\in A^N} p(W)\log_2 p(W) = \sum_{W\in O}\frac{1}{\#O}\log_2 \#O = \log_2 \#O\,.$$

Die Entropie pro Buchstabe des Alphabets A ist damit gleich

$$\frac{\log_2 \#O}{N}\,.$$

Diese Entropie pro Buchstabe können wir berechnen; für die Quellen aus §1 haben wir dies dort bereits getan: Geht man nur von den Buchstabenhäufigkeiten aus, ist sie gleich 4,04088 Bit, und bei Berücksichtigung der Kontakthäufigkeiten gleich

$$\frac{4{,}04088 + 3{,}39765(N-1)}{N}\ \text{Bit}\,.$$

Also ist im ersten Fall

$$\#O \approx 2^{4{,}04088N}$$

und im zweiten

$$\#O \approx 2^{4{,}04088+3{,}39765(N-1)}\,.$$

In A^N gibt es insgesamt

$$26^N \approx 2^{4{,}7044N}$$

Nachrichten; das Verhältnis $\#O$ zu $\#A^N$ läßt sich daher schreiben als

$$\frac{\#O}{\#A^N} = 2^{-R_N}\,,$$

wobei im ersten Fall

$$R_N \approx (4{,}7044 - 4{,}04088)N = 0{,}66352N$$

ist und im zweiten

$$R_N \approx 4{,}7044N - 4{,}04088 + 3{,}39765(N-1) = 1{,}30675N - 0{,}64323\,.$$

Damit können wir uns daran machen, die Mehrdeutigkeit eines Schlüssels für diese (und viele andere) Quellen abzuschätzen: Wir ersetzen die jeweilige Quelle durch ein „durchschnittliche“ ergodische Quelle derselben Entropie und rechnen mit dieser in der Hoffnung, daß wir dadurch keinen allzu großen Fehler machen. Außerdem nehmen wir

an, daß N hinreichend groß sei, so daß wir den obigen Satz anwenden können.

Nach Definition ist die Mehrdeutigkeit des Schlüssels nach Empfang einer Chiffretext-Nachricht C der Länge N gleich

$$H_{S|Ch} \underset{\text{def}}{=} - \sum_{(C,s) \in A^N \times S} p_{Ch,S}(C,s) \log_2 p_{S|Ch}(s|C)$$

Wenn wir davon ausgehen, daß jeder Schlüssel mit derselben Wahrscheinlichkeit auftritt, ist für jeden Schlüssel $s \in S$

$$p_S(s) = \frac{1}{\#S} .$$

und für einen Klartext $W \in A^N$ ist

$$p_K(W) = \begin{cases} 1/\#O & \text{falls } W \in O \\ 0 & \text{sonst} \end{cases} ;$$

eine Nachricht $C \in A^N$ kann also nur dann als Chiffretext auftreten, wenn es (mindestens) einen Schlüssel $s \in S$ gibt, so daß der zugehörige Klartext $W = T_s^{-1}(C)$ in O liegt. Wenn wir die Anzahl aller solcher Schlüssel mit $N_S(C)$ bezeichnen, ist also die Wahrscheinlichkeit des Chiffretexts C gleich

$$p_{Ch}(C) = \frac{N_S(C)}{\#S} \cdot \frac{1}{\#O} = \frac{N_S(C)}{\#S \cdot \#O} ;$$

die Wahrscheinlichkeit für Chiffretext C verschlüsselt mit $s \in S$ ist

$$p_{Ch,S}(C,s) = \begin{cases} \frac{1}{\#O \cdot \#S} & \text{falls } T_s^{-1}(C) \in O \\ 0 & \text{sonst} \end{cases} ,$$

und damit ist die bedingte Wahrscheinlichkeit für den Schlüssel s nach Empfang des Chiffretexts C

$$p_{S|Ch}(s|C) = \frac{p_{Ch,S}(C,s)}{p_{Ch}(C)} = \frac{1}{N_S(C)} .$$

Somit ist

$$\begin{aligned} H_{S|Ch} &= -\sum_{(C,s)\,\in\, A^N \times S} p_{Ch,S}(C,s)\log_2 p_{S|Ch}(s|C) \\ &= -\sum_{C\in A^N}\sum_{\substack{s\in S\\ T_s^{-1}(C)\in O}} \frac{1}{\#O\cdot\#S}(-\log_2 N_S(C)) \\ &= \sum_{C\in A^N}\frac{N_S(C)}{\#O\cdot\#S}\log_2 N_S(C)\,. \end{aligned}$$

Diese Summe können wir nicht weiter ausrechnen, da wir die Zahlen $N_S(C)$ nicht kennen. Nun haben wir aber angenommen, daß wir ein *durchschnittliches* Kryptosystem haben, d.h. wir interessieren uns für den Mittelwert von $H_{S|Ch}$ über *alle* Kryptosysteme, und darüber können wir Aussagen machen:

Für einen zufällig gewählen Text $C \in A^N$ und einen zufällig gewählten Schlüssel $s \in S$ ist $T_s^{-1}(C) \in O$ mit Wahrscheinlichkeit

$$\frac{\#O}{\#A^N} = \frac{\#O}{n^N} = 2^{-R_N}\,,$$

wobei die reelle Zahl R_N wie oben so gewählt wird, daß die rechtsstehende Gleichung gilt. Entsprechend ist die Wahrscheinlichkeit dafür, daß $T_s^{-1}(C)$ nicht in O liegt, gleich

$$1 - 2^{-R_N}\,.$$

Die Wahrscheinlichkeit dafür, daß $N_S(C)$ gleich einer festen Zahl m ist, können wir interpretieren als die Wahrscheinlichkeit dafür, daß für m Schlüssel s gilt $T_s^{-1}(C) \in O$, während für die restlichen $\#S - m$ Schlüssel gilt $T_s^{-1}(C) \notin O$. Falls die m Schlüssel vorgegeben sind, ist diese Wahrscheinlichkeit gleich

$$\left(2^{-R_N}\right)^m \left(1 - 2^{-R_N}\right)^{\#S-m}\,,$$

und da es $\binom{\#S}{m}$ Möglichkeiten gibt, aus $\#S$ Schlüsseln m auszuwählen, ist die Wahrscheinlichkeit dafür, daß $N_S(C) = m$ ist, gleich

$$\binom{\#S}{m}\left(2^{-R_N}\right)^m \left(1 - 2^{-R_N}\right)^{\#S-m}\,,$$

die Anzahl der Chiffretexte C mit $N_S(C) = m$ ist also gleich

$$\binom{\#S}{m} \left(2^{-R_N}\right)^m \left(1 - 2^{-R_N}\right)^{\#S-m} \#A^N \,.$$

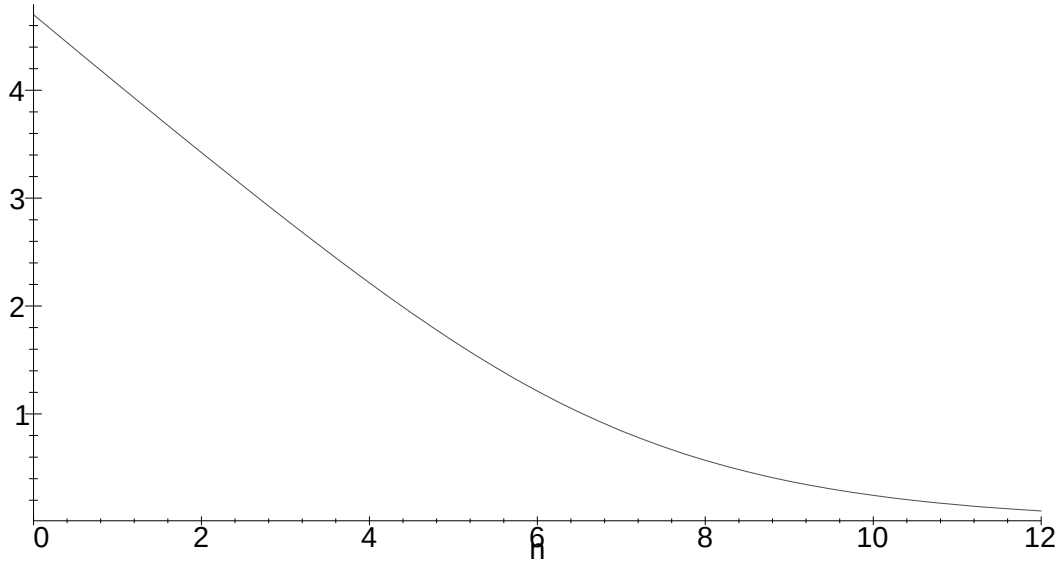
Der durchschnittliche Wert von $H_{S|Ch}$ berechnet sich somit zu

$$\begin{aligned}
&\sum_{m=0}^{\#S} \binom{\#S}{m} \left(2^{-R_N}\right)^m \left(1 - 2^{-R_N}\right)^{\#S-m} \#A^N \frac{m}{\#O\#S} \log_2 m \\
&= \frac{\#A^N}{\#O} \frac{1}{\#S} \sum_{m=0}^{\#S} \binom{\#S}{m} \left(2^{-R_N}\right)^m \left(1 - 2^{-R_N}\right)^{\#S-m} m \log_2 m \\
&= \frac{2^{R_N}}{\#S} \sum_{m=0}^{\#S} \binom{\#S}{m} \left(2^{-R_N}\right)^m \left(1 - 2^{-R_N}\right)^{\#S-m} m \log_2 m \,.
\end{aligned}$$

Falls wir $\#S = 26$ setzen und von den beiden oben betrachteten Wahrscheinlichkeitsmodellen ausgehen, wenn wir also ein *zufälliges* Kryptosystem betrachten, das dieselben Parameter hat wie das System der 26 CAESAR-Substitutionen, läßt sich dies leicht berechnen; das Ergebnis ist für die beiden oben betrachteten Wahrscheinlichkeitsmodelle in den beiden folgenden Abbildungen dargestellt. Wie man sieht, ist die Mehrdeutigkeit des Schlüssels ab etwa zehn bis zwölf *bzw.* sechs bis acht Buchstaben Chiffretext praktisch gleich null, was auch ungefähr unseren Experimenten mit der CAESAR-Chiffre entspricht.

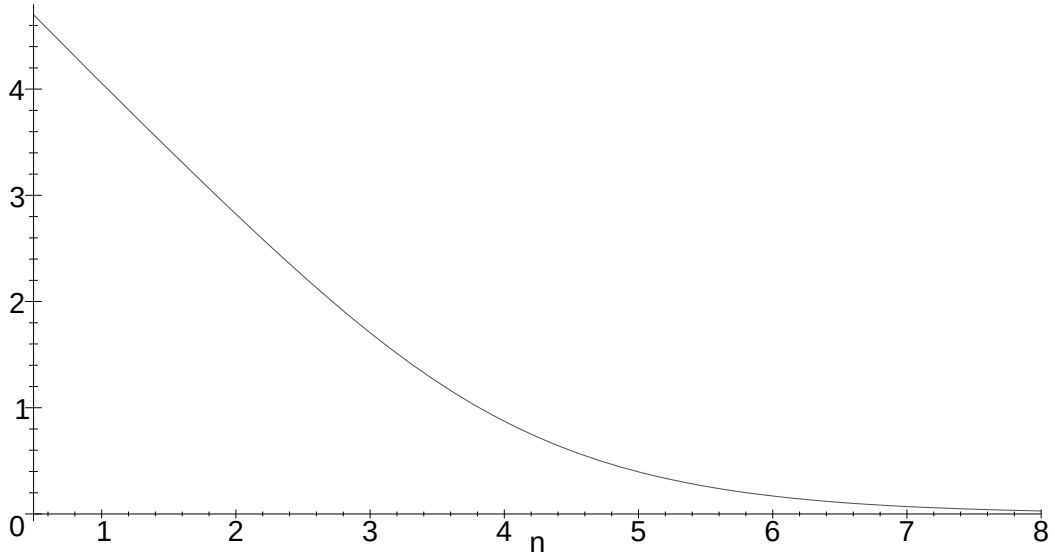
Wie man den Kurven ansieht, nimmt der Mehrdeutigkeit des Schlüssels anfänglich etwa linear ab mit n. Wer die Binomialverteilung kennt, sieht auch leicht, warum dies so ist: Falls der Erwartungswert $2^{-R_N}\#S$ von $N_S(C)$ hinreichend groß ist, sind alle einigermaßen wahrscheinliche Werte in seiner näheren Umgebung, man kann daher ohne allzu großen Fehler $N_S(C)$ durch diesen Wert ersetzen und erhält

$$\begin{aligned}
H_{S|Ch} &= \sum_{C \in A^N} \frac{N_S(C)}{\#O \cdot \#S} \log_2 N_S(C) \\
&\approx \#A^N \cdot \frac{2^{-R_N}\#S}{\#O \cdot \#S}(-R_N + \log_2 \#S) \\
&= \log_2 \#S - R_N \,.
\end{aligned}$$



Mehrdeutigkeit des Schlüssels via Buchstabenhäufigkeiten

Anfänglich gewinnt man also aus N Buchstaben Chiffretext etwa R_N Bit des Schlüssels. Falls R_N wie in unseren Beispielen eine lineare Funktion von N ist, nimmt die Mehrdeutigkeit des Schlüssels also anfänglich linear ab.



Mehrdeutigkeit des Schlüssels via Kontakthäufigkeiten

Diese Rechnungen zeigen einmal mehr, daß ein Gegner vor allem von der Redundanz R_N der Nachrichtenquelle profitieren kann; je kleiner diese ist, desto weniger Information kann er gewinnen. Durch vorherige Datenkomprimierung läßt sich daher die Sicherheit der meisten Kryptosysteme deutlich erhöhen.

§6: Asymptotische Gleichverteilung

Wie wir am Ende des zweiten Paragraphen gesehen haben, kann die Entropie einer Quelle gelegentlich interpretiert werden als die mittlere Anzahl von Bit, die wir zur Kodierung eines Buchstabens benötigen. Dies funktioniert aber nicht immer: Bei einer Quelle, die drei Buchstaben mit gleicher Wahrscheinlichkeit produziert, kann es natürlich keine Kodierung geben, bei der wir im Durchschnitt genau $\log_2 3$ Bit brauchen – der mittlere Aufwand läßt sich bei jeder Kodierung als Bruch mit Nenner drei darstellen. Unsere beste Wahl besteht darin, daß wir einen der Buchstaben etwa als die Null darstellen und die beiden anderen als 10 und 11; der mittlere Aufwand beträgt dann $5/3$ Bit.

Wenn wir je zwei Buchstaben zu einer Gruppe zusammenfassen, haben wir neun Paare, die jeweils mit Wahrscheinlichkeit $1/9$ auftreten – falls wir annehmen, daß unsere Quelle Buchstaben jeweils unabhängig vom Vorgänger produziert. Kodieren wir die ersten vier Paare durch $000, 001, 010$ und 011, so können wir die restlichen fünf beispielsweise darstellen als $1000, 1001, 1010, 1011$ und 1100; der mittlere Aufwand ist also gesunken auf

$$\frac{4}{9} \times 3 + \frac{5}{9} \times 4 = \frac{32}{9} \text{ Bit}$$

pro Paar oder $16/9$ Bit pro Buchstabe. Bei Blöcken von fünf Buchstaben haben wir $3^5 = 243$ verschiedene Blöcke; indem wir einfach die Zahlen von 0 bis 242 im Zweiersystem darstellen, kommen wir also mit acht Bit aus (tatsächlich sogar geringfügig weniger, da wir noch ein paar 7-Bit-Kodierungen vergeben können) und sind damit bei knapp 1,6 Bit pro Buchstabe angelangt, was bereits recht nahe bei $\log_2 3 \approx 1{,}585$ angelangt.

Wir erwarten, daß wir durch Übergang zu immer größeren Blöcken dem Wert $\log_2 3$ immer näher kommen, auch wenn wir ihn zumindest in diesem Beispiel nie erreichen können: Wie man zeigen kann, ist $\log_2 3$ eine transzendente, insbesondere also irrationale Zahl, und der Aufwand pro Buchstabe ist bei dieser Quelle unabhängig von der Kodierung stets eine rationale Zahl.

Wir müssen uns daher begnügen mit einer Näherungsaussage.

Betrachten wir als Beispiel eine Folge voneinander unabhängiger Zufallsvariablen $X_1, X_2, \ldots$ mit einem Alphabet $\{0, 1\}$ aus zwei Buchstaben. Jede Variable X_i möge mit Wahrscheinlichkeit p eine Eins liefern und dementsprechend mit Wahrscheinlichkeit $q = 1 - p$ eine Null. Die Entropie ist dann jeweils $H(X_i) = -p \log_2 p - q \log_2 q$.

Das n-tupel $(X_1, \ldots, X_n)$ produziert Blöcke aus n Binärziffern; Erwartungswert für die Anzahl der Einsen in so einem Block ist pn. Um eine grobe Näherung für die Wahrscheinlichkeit eines *typischen* Blocks zu bekommen, nehmen wir erstens an, pn sei eine ganze Zahl, und zweitens, daß in einem typischen Block genau pn Einsen auftreten. Die Wahrscheinlichkeit eines solchen typischen Blocks ist dann

$$p^{np} q^{n-np} = p^{np} q^{nq} = 2^{np \log_2 p + nq \log_2 q} = 2^{-H(X_i)} .$$

Natürlich gibt es (außer im Fall $p = \frac{1}{2}$) auch Blöcke, deren Wahrscheinlichkeit deutlich von diesem Wert abweicht, zum Beispiel die beiden, die aus lauter Nullen *bzw.* lauter Einsen bestehen, aber nach dem Gesetz der großen Zahl sollte für hinreichend große n die Wahrscheinlichkeit einer großeren Abweichung von diesem Wert sehr gering sein.

Dieser Philosophie entsprechend definieren wir nun für Zufallsvariablen mit beliebiger Verteilung eine *typische Menge* und untersuchen deren Eigenschaften:

Definition: $X_1, X_2, \ldots$ sei eine Folge voneinander unabhängiger Zufallsvariablen mit Werten in einem Alphabet A, die allesamt die gleiche Wahrscheinlichkeitsverteilung haben. Für ein Tupel $(a_1, \ldots, a_n) \in A^n$ bezeichne

$$p(a_1, \ldots, a_n) = \prod_{i=1}^{n} p(a_i)$$

die Wahrscheinlichkeit dafür, daß für $i = 1, \ldots, n$ die Zufallsvariable X_i den Wert a_i annehme. Bezeichnet

$$H = -\sum_{x \in A} p(x) \log_2 p(x)$$

die Entropie der Zufallsvariablen X_i, definieren wir jedes $\varepsilon > 0$ eine *typische Menge*

$$A_\varepsilon^n = \left\{ (a_1, \ldots, a_n) \in A^n \;\middle|\; 2^{-n(H+\varepsilon)} \leq p(x_1, \ldots, a_n) \leq 2^{-n(H-\varepsilon)} \right\}.$$

Einige wichtige Eigenschaften dieser Menge sind im folgenden Satz zusammengefaßt:

Satz: $X_1, X_2, \ldots$ sei eine Folge voneinander unabhängiger identisch verteilter Zufallsvariablen mit Werten in einem Alphabet A; ihre Entropie sei H. Dann gilt:

a) Für alle $(a_1, \ldots, a_n) \in A_\varepsilon^n$ ist $\left| -\frac{1}{n} \log_2 p(a_1, \ldots, a_n) - H \right| < \varepsilon$.

b) Für hinreichend große Werte von n ist die Wahrscheinlichkeit dafür, daß ein Element von A^n in A_ε^n liegt, größer als $1 - \varepsilon$.

c) Die Kardinalität $\#A_\varepsilon^n$ von A_ε^n ist höchstens gleich $2^{n(H+\varepsilon)}$.

d) Für hinreichend große n ist $\#A_\varepsilon^n \geq (1 - \varepsilon) 2^{n(H+\varepsilon)}$.

Beweis: a) folgt sofort aus der Definition der typischen Menge, wenn wir in der definierenden Ungleichung zu Logarithmen übergehen und durch n dividieren.

Für *b)* erinnern wir uns an das (schwache) Gesetz der großen Zahlen: Danach konvergieren die Mittelwerte $\frac{1}{n}(Y_1 + \cdots + Y_n)$ einer Folge voneinander unabhängiger aber identisch verteilter reellwertiger Zufallsvariablen für $n \to \infty$ stochastisch gegen den gemeinsamen Erwartungswert der Y_i. Die Zufallsvariablen Y_i definieren wir für diesen Beweis wie folgt: Wenn X_i den Wert $a \in A$ liefert, soll Y_i den Wert $-\log_2 p(a)$ liefern. Der gemeinsame Erwartungswert der Y_i ist dann die Entropie

$$H = -\sum_{a \in A} p(a) \log_2 p(a)$$

der X_i; es gibt daher zu jedem $\delta > 0$ ein $N \in \mathbb{N}$, so daß die Wahrscheinlichkeit des Ereignisses $\left| -\frac{1}{n} \log_2 p(a_1, \ldots, a_n) - H \right| < \varepsilon$ größer

ist als $1-\delta$ für alle $n \geq N$. Speziell können wir auch für $\delta = \varepsilon$ so ein N finden, womit *b)* bewiesen wäre.

c) folgt durch eine einfache Abschätzung: Da

$$\begin{aligned} 1 &= \sum_{(a_1,\ldots,a_n)\in A^n} p(a_1,\ldots,a_n) \geq \sum_{(a_1,\ldots,a_n)\in A_\varepsilon^n} p(a_1,\ldots,a_n) \\ &\geq \sum_{(a_1,\ldots,a_n)\in A_\varepsilon^n} 2^{-n(H+\varepsilon)} = 2^{-n(H+\varepsilon)} \# A_\varepsilon^n \end{aligned}$$

ist, muß $\# A_\varepsilon^n \leq 2^{n(H+\varepsilon)}$ sein.

Zum Beweis von *d)* schließlich schätzen wir in umgekehrter Richtung ab, ausgehend von Aussage *b):* Für alle hinreichend großen n ist

$$1-\varepsilon < \sum_{(a_1,\ldots,a_n)\in A_\varepsilon^n} p(a_1,\ldots,a_n) \leq \sum_{(a_1,\ldots,a_n)\in A_\varepsilon^n} 2^{-n(H-\varepsilon)} = 2^{-n(H-\varepsilon)} \# A_\varepsilon^n$$

und damit $\# A_\varepsilon^n \geq (1-\varepsilon) 2^{n(H-\varepsilon)}$.

■

Als erste Anwendung können wir abschätzen, wie viele Bit wir brauchen, wenn wir in der vorliegenden Situation blockweise kodieren: Wenn wir zu vorgegebenem $\varepsilon > 0$ die Blocklänge n so groß wählen, daß *b)* erfüllt ist, haben wir einerseits höchstens $2^{n(H+\varepsilon)}$ Elemente in A_ε^n und kommen daher mit $n(H+\varepsilon)$ Bit pro Block aus, wenn wir nur diese Blöcke kodieren. Die Wahrscheinlichkeit dafür, daß einer der restlichen Blöcke auftritt, ist kleiner als ε; auch wenn wir für die Kodierung solcher Blöcke erheblich längere Bitfolgen ansetzen müssen, beispielsweise solche der Länge $n \log_2 m$, wobei m die Elementanzahl des Alphabets bezeichnet, wird diese Länge mit ε multipliziert, so daß wir im Mittel nur

$$n(H+\varepsilon) + \varepsilon \cdot n \log_2 m = n\big(H + \varepsilon(1+\log_2 m)\big)$$

brauchen; pro Zeichen sind das

$$H + \varepsilon(1+\log_2 m) \text{ Bit.}$$

Mit hinreichend großen Blocklängen kommen wir somit beim mittleren Kodierungsaufwand in der Tat beliebig nahe an die Entropie heran.

§7: Datenkompression

Wir haben schon mehrfach gesehen, daß die Entropie einer Zufallsvariablen in einfachen Fällen interpretiert werden kann als die mittlere Bitanzahl für eine Kodierung ihres Alphabets. Diesen Zusammenhang und seine Anwendung auf die Komprimierung verschiedener Arten von Daten soll in diesem Paragraphen untersucht werden.

Zur Speicherung oder Übermittlung der von einer Zufallsvariablen oder einem stochastischen Prozess produzierten Daten müssen wir die Elemente des Alphabets A in geeigneter Weise kodieren mit Zeichenfolgen, die durch die verwendete Technik vorgegeben sind; heutzutage sind dies meist Bitfolgen. Wir gehen allgemein aus von einem Zeichensatz, der zwar meist, aber nicht immer, gleich der Menge $\{0, 1\}$ sein wird. Historisch bedeutsame Beispiele von Mengen C mit mehr als zwei Elementen sind etwa die Morsezeichen mit $C = \{\text{kurz, lang, Pause}\}$ oder die in der Seefahrt gebräuchlichen Flaggenalphabete.

Ziel der Datenkompression ist es, die Anzahl der Zeichen für die betrachteten Daten möglichst klein zu halten; bei einem verlustfreien Komprimierungsverfahren sollen aber die ursprünglichen Daten trotzdem noch exakt rekonstruierbar sein.

Nicht alle in der Praxis verwendeten Verfahren sind verlustfrei; beim mp3-Verfahren etwa wird ausgenutzt, daß manche Frequenzen unser Ohr für andere Frequenzen blockieren, so daß man diese aus dem Signal herausfiltern kann. Auch das JPEG-Verfahren, mit dem sich der letzte Abschnitt dieses Paragraphen beschäftigt, ist oft nicht verlustfrei; hier gibt es einen Qualitätsfaktor als Parameter, der die tolerierbaren Verluste quantifiziert.

Es ist klar, daß es keinen universellen Algorithmus zur Datenkompression geben kann: Gäbe es nämlich ein Verfahren, das für beliebige Dateien einen Kompressionsfaktor $\alpha < 1$ garantieren würde, so könnte man dieses Verfahren iterativ anwenden und nach n Anwendungen eine Kompressionsrate von α^n erreichen. Bei hinreichend großem n könnte man daher jede Datei auf weniger als ein Bit komprimieren, was natürlich absurd ist.

Ein Kompressionsverfahren kann also nur auf Dateien mit spezieller Struktur erfolgreich angewandt werden. Wir werden in diesem Paragraphen zwei Ansätze diskutieren: Die Entropiekodierung, bei der die Unterschiede zwischen den Häufigkeiten der einzelnen Buchstaben ausgenutzt wird, und die Datenkomprimierung durch lineare Transformationen, die bei einem stochastischen Prozess eine weitestgehende Dekorrelation der beteiligten Zufallsvariablen erreichen wollen.

a) Quellenkodierung

Wir gehen zunächst aus vom einfachsten Fall einer einzigen Zufallsvariablen X. Sie nehme Werte an in einem Alphabet $A = \{a_1, \ldots, a_m\}$, wobei der Buchstabe a_i mit Wahrscheinlichkeit p_i auftrete.

Zur Speicherung oder Übermittlung der von X produzierten Daten müssen wir die Elemente von A in geeigneter Weise kodieren mit Zeichenfolgen, die durch die verwendete Technik vorgegeben sind; heutzutage sind dies meist Bitfolgen. Wir gehen allgemein aus von einem Zeichensatz $\mathcal{D}$, der zwar meist, aber nicht immer, gleich der Menge $\{0, 1\}$ sein wird. Die Elementanzahl von $\mathcal{D}$ bezeichnen wir mit D.

Das Wort *Kodierung* hat in der Informationstheorie mehrere deutlich verschiedene Bedeutungen: Wir haben einmal Codes, die dazu dienen allfällige Lese- und Übertragungsfehler zu erkennen und teilweise auch zu korrigieren; diese fehlererkennenden und fehlererkorrigierenden Codes bilden den Inhalt mathematischer Vorlesungen über *Kodierungstheorie;* Informationstechniker reden hier von *Kanalkodierung.* Dann gibt es schon seit mindestens zweieinhalb Jahrtausenden *Geheimcodes,* mit denen Text so verschlüsselt werden soll, daß ihn ein Unbefugter nicht rekonstruieren kann; diese werden in Vorlesungen über Kryptologie (oder, wenn die reine Verschlüsselung im Vordergrund steht, auch Kryptographie) behandelt. Die Kodierung, mit der wir uns hier beschäftigen, greift bereits eine Stufe vor diesen beiden und heißt *Quellenkodierung;* sie wird oft zusammen mit Kanalkodierung und/oder Verschlüsselung eingesetzt.

Definition: Ein *Quellencode* für eine Zufallsvariable X mit Werten in einem Alphabet A ist eine Abbildung C, die jedem Element $x \in A$

eine Folge von Elementen aus $\mathcal{D}$ zuordnet. Das Codewort zu $x \in A$ bezeichnen wir mit $C(x)$, seine Länge mit $\ell(x)$. Die *mittlere Länge* $L(C)$ ist der Erwartungswert

$$\mathbb{E}\big(\ell(X)\big) = \sum_{x \in A} p(x)\ell(x)\,.$$

Die wohl bekanntesten Beispiele von Quellencodes sind der ASCII-Code *(American Standard Code for Information Interchange),* der ein Alphabet aus 128 Zeichen durch Folgen aus je sieben Bit darstellt, denen zwecks Fehlererkennung praktisch immer ein Paritätsbit angehängt wird, sowie seine Erweiterungen wie ISO Latin-1, die ASCII zu einem echten Achtbitcode erweitern, in dem auch Umlaute und Ähnliches zum Alphabet gehören. Vor allem im *World Wide Web* wird oft auch der sogenannte *Unicode* verwendet, der mit 17 Ebenen zu je 16 Bit jedem irgendwo auf der Welt benutzten Schriftzeichen eine digitale Entsprechung zuordnen will.

Laut obiger Definition ist ein Quellencode einfach irgendeine Abbildung; wenn diese nicht injektiv ist, reden wir von einem *singulären* Code. Da solche Codes nur selten nützlich sind, werden wir uns im Folgenden auf nichtsinguläre Codes beschränken.

Auch bei diesen kann es aber noch Probleme mit der eindeutigen Rekonstruierbarkeit geben wenn wir uns bei der Kodierung nicht auf einzelne Buchstaben beschränken, sondern – wie dies wohl meist der Fall sein wird – Buchstaben*folgen* betrachten: Kodieren wir etwa die 26 Buchstaben des Alphabets durch die im Zweiersystem geschriebenen Zahlen von 0 bis 25, so ist diese Abbildung sicherlich injektiv; setzen wir aber die Codes $C(D) = 11, C(A) = 0$ und $C(S) = 10010$ einfach hintereinander, können wir das Ergebnis auch beispielsweise als GEC lesen statt als DAS.

Um dieses Problem zu vermeiden, könnten wir uns auf Codes beschränken, bei denen alle Codewörter $C(x)$ dieselbe Länge haben, wie es beispielsweise bei ASCII der Fall ist oder auch bei den heute kaum noch benutzten Fernschreibern. Wenn allerdings die Zeichen des Alphabets (wie etwa im Fall der Buchstaben eines deutschen Texts) deutlich verschiedene Wahrscheinlichkeiten haben, können wir die mittlere Länge

des Codes oft drastisch reduzieren, wenn wir den häufigen Buchstaben kurze Codewörter zuordnen. Die geschieht beispielsweise beim Morse-Code, der mit den drei Zeichen *kurz, lang* und *Pause* arbeitet; hier wird das E durch einmal kurz kodiert, das N durch einmal lang, das Y aber durch lang, kurz, lang, lang. Zum Trennen der einzelnen Buchstaben dient das dritte Zeichen, die Pause.

Wir interessieren uns für Codes variabler Länge, bei denen die eindeutige Dekodierbarkeit ohne spezielles Trennzeichen gewährleistet ist:

Definition: *a)* Ein Code heißt *eindeutig dekodierbar,* wenn es keine zwei Folgen von Buchstaben aus dem Alphabet A gibt, die auf dieselbe Zeichenfolge abgebildet werden.
b) Ein Code heißt *Praefixcode,* wenn es keine zwei Buchstaben $x, y \in A$ gibt, für die $C(x)$ mit den ersten $\ell(x)$ Zeichen von $C(y)$ übereinstimmt.

Offensichtlich ist jeder Praefixcode eindeutig dekodierbar; umgekehrt ist jedoch nicht jeder eindeutig dekodierbare Code ein Praefixcode: Kodieren wir etwa ein dreielementiges Alphabet durch die Vorschrift $C(a) = 0, C(b) = 001$ und $C(c) = 11$ und sehen beim Dekodieren als erstes Zeichen eine Eins, so kann der nächste Buchstabe nur ein b sein. Sehen wir eine Folge von n Nullen, gefolgt von einer geraden Anzahl von Einsen, so müssen wir n mal den Buchstaben a haben, gefolgt von einem c; folgt aber eine ungerade Anzahl von Einsen, so haben wir $n-2$ mal den Buchstaben a, gefolgt von einem b. Somit ist C eindeutig dekodierbar, obwohl $C(a)$ ein Praefix von $C(b)$ ist.

Die eindeutige Dekodierbarkeit schränkt die Anzahl möglicher Codewörter einer vorgegebenen Länge ein; insbesondere gilt die folgende

Ungleichung von Kraft und McMillan: Ist C ein eindeutig dekodierbarer Code für das Alphabet A und bezeichnet D die Anzahl der Codezeichen, so ist

$$\sum_{x \in A} D^{-\ell(x)} \leq 1 \,.$$

Beweis: Für jede natürliche Zahl k läßt sich C fortsetzen zu einem Code für das Alphabet A^k, indem wir einem k-tupel $(x_1, \ldots, x_k)$ von

Buchstaben die hintereinander gesetzten Codewörter $C(x_1), \ldots, C(x_k)$ zuordnen, aufgefaßt als ein Codewort der Länge $\ell(x_1) + \cdots + \ell(x_k)$. Wegen der eindeutigen Dekodierbarkeit von C ist auch dieser erweiterte Code nichtsingulär.

Nach dem Distributivgesetz ist

$$\left(\sum_{x\in A} D^{-\ell(x)}\right)^k = \sum_{(x_1,\ldots,x_k)\in A^k} D^{-\ell(x_1)} \cdots D^{-\ell(x_k)} = \sum_{\mathbf{x}\in A^k} D^{-\ell(\mathbf{x})} .$$

Bezeichnet $a(n)$ die Anzahl der k-tupel $\mathbf{x} \in A^k$, denen ein Codewort der Länge n zugeordnet wird, können wir dies auch schreiben als

$$\sum_{n=1}^{k\ell_{\max}} a(n) D^{-n} ,$$

wobei $\ell_{\max}$ das Maximum aller $\ell(x)$ für $x \in A$ bezeichnet. Da der Code nichtsingulär ist, kann $a(n)$ nicht größer sein als die Anzahl D^n aller möglicher Codewörter der Länge n; somit ist

$$\left(\sum_{x\in A} D^{-\ell(x)}\right)^k \le \sum_{n=1}^{k\ell_{\max}} D^n D^{-n} = k\ell_{\max}$$

und damit

$$\sum_{x\in A} D^{-\ell(x)} \le \sqrt[k]{k\ell_{\max}} .$$

Dies gilt für alle k, und wegen

$$\log \sqrt[k]{k\ell_{\max}} = \frac{\log k}{k} + \frac{\log \ell_{\max}}{k}$$

folgt (z.B. mit der Regel von DE L'HÔPITAL)

$$\lim_{k\to\infty} \sqrt[k]{k\ell_{\max}} = 1 ,$$

was die behauptete Ungleichung beweist. ∎

Dieser Satz wurde 1949 von LEON G. KRAFT im Rahmen seiner Master Thesis im Fach Elektrotechnik am MIT für Praefixcodes bewiesen. Für beliebige eindeutig dekodierbare Codes bewies sie BROCKWAY MCMILLAN von den Bell Telephone Labs unabhängig von

Kraft 1956 in seiner Arbeit *Two Inequalities Implied by Unique Decipherability* in den IEEE Transaction on Information Theory, Band 2(4), S. 115-116.

Umgekehrt gilt

Satz: Ist $\ell\colon A \to \mathbb{N}$ eine Abbildung des Alphabets A in die natürlichen Zahlen mit

$$\sum_{x \in A} D^{-\ell(x)} \leq 1\,,$$

so gibt es einen Praefixcode C mit D-elementigem Zeichensatz, für den das Codewort $C(x)$ die Länge $\ell(x)$ hat.

Beweis: Die D Zeichen, aus denen die Codewörter gebildet werden, seien $z_1, \ldots, z_D$, und ℓ_{max} sei das Maximum der Längen $\ell(x)$; das zu kodierende Alphabet sei $A = \{a_1, \ldots, a_m\}$.

Wir zeichnen einen Baum, indem wir ausgehend von einem festen Punkt, der Wurzel, D gerichtete Strecken zeichnen, die wir mit z_1 bis z_D markieren. Vom Endpunkt einer jeden dieser Strecken zeichnen wir wieder D solche Strecken und so weiter, bis wir Streckenzüge der Länge ℓ_{max} haben. Punkte, die vom Anfangspunkt aus über einen Streckenzug der Länge n erreichbar sind, bezeichnen wir als Knoten der Tiefe n. Jeder Knoten ist eindeutig beschreibbar durch die Folge der Markierungen $z_{i_1} \ldots z_{i_n}$ der Strecken, die zu ihm führen; Knoten der Tiefe n entsprechen also potentiellen Codewörtern der Länge n. Diese ordnen wir lexikographisch durch die Übereinkunft, daß z_i vor z_j kommen soll, wenn $i < j$ ist.

Zur Konstruktion des gewünschten Codes ordnen wir als erstes dem Buchstaben a_1 das Codewort zu, das aus $\ell(a_1)$ Zeichen z_1 besteht. Danach entfernen wir den zu diesem Codewort gehörenden Knoten der Tiefe n aus dem Baum sowie alle Knoten, die über diesen Punkt erreichbar sind – sie entsprechen schließlich Codewörtern, die das gerade vergebene Codewort als Anfang hätten.

Wenn wir Codewörter für a_1 bis a_{r-1} vergeben haben, konstruieren wir das Codewort $C(a_r)$ wie folgt: Unter allen noch verbleibenden Knoten der Tiefe $\ell(a_r)$ nehmen wir den lexikographisch ersten und definieren $C(a_r)$ als das dazugehörige Codewort; sodann streichen wir

wieder sowohl diesen Knoten als auch alle über ihn erreichbaren Knoten größerer Tiefe.

Das kann natürlich nur dann funktionieren, wenn es noch einen Knoten der Tiefe $\ell(a_r)$ gibt. Dazu genügt es, wenn wir zeigen, daß es noch mindestens einen Knoten der Tiefe $\ell_{\max}$ gibt, denn der Weg zu einem solchen Knoten führt durch Knoten aller kleineren Tiefen.

Wenn wir ein Codewort der Länge n vergeben, streichen wir mit dem zugehörigen Knoten der Tiefe n auch alle über diesen Knoten erreichbare tiefere; in Tiefe $\ell_{\max}$ sind das $D^{\ell\max - n}$ Stück. Durch die Vergabe von Codewörtern für a_1 bis a_{r-1} wurden somit

$$\sum_{i=1}^{r-1} D^{\ell\max - \ell(a_i)} = D^{\ell\max} \sum_{i=1}^{r-1} D^{-\ell(a_i)}$$

Knoten gelöscht. Anfangs gab es zu jedem n nach Konstruktion D^n Knoten der Tiefe n, also $D^{\ell\max}$ Knoten der maximalen Tiefe. Nach Voraussetzung ist für $r \leq n$

$$\sum_{i=1}^{r-1} D^{-\ell(a_i)} < \sum_{x \in A} D^{-\ell(x)} \leq 1\,,$$

die Anzahl bereits gestrichener Knoten maximaler Tiefe ist also echt kleiner als die ursprünglich vorhandene Anzahl.

■

Zusammen zeigen die beiden gerade bewiesenen Sätze, daß es zu jedem eindeutig dekodierbaren Code einen Praefixcode gibt, dessen Codewörter exakt dieselbe Länge haben. Auf der Suche nach möglichst guten Codes können und werden wir uns daher auf Praefixcodes beschränken.

b) Optimale Codes

Da sowohl die Speicherung als auch die Übertragung von Daten oftmals knappe Ressourcen in Anspruch nehmen wie beispielsweise den Speicher einer kleinen Digitalkamera oder die Kapazität eines zumindest zeitweise sehr stark belasteten Mobilfunknetzes, ist es für viele Anwendungen wichtig, den benötigten Aufwand auf das unbedingt notwendige

Minimum zu beschränken. Auf dem Niveau der Quellenkodierung bedeutet dies insbesondere, daß die mittlere Länge des verwendeten Codes möglichst klein sein soll.

Sei also X eine Zufallsvariable mit Werten in einem m-elementigen Alphabet A, dessen i-tes Element mit Wahrscheinlichkeit p_i angenommen werde. Wir suchen einen Code, dessen mittlere Länge

$$L = \sum_{i=1}^{m} p_i \ell_i$$

minimal ist; dabei steht ℓ_i für die Länge des Codeworts zum i-ten Element des Alphabets.

Wie wir gerade gesehen haben, müssen die Längen der Codewörter eines eindeutig dekodierbaren Quellencodes die Ungleichung von KRAFT und MCMILLAN erfüllen; umgekehrt gibt es auch zu jeder Längenverteilung, die diese Ungleichung erfüllt, einen Praefixcode. Somit müssen wir L minimieren unter der Nebenbedingung

$$\sum_{i=1}^{m} D^{-\ell_i} \leq 1 \,.$$

Wenn wir für den Augenblick vergessen, daß die ℓ_i natürliche Zahlen sein müssen, haben wir hier ein klassisches Extremwertproblem mit Nebenbedingung, das wir mit Hilfe eines LAGRANGE-Multiplikators lösen können: Da eine lineare Funktion keine lokalen Extrema hat, muß die Nebenbedingung für alle Extrema eine Gleichung sein, und somit müssen

$$\operatorname{grad} L = \begin{pmatrix} p_1 \\ \vdots \\ p_m \end{pmatrix} \quad \text{und} \quad \operatorname{grad} \sum_{i=1}^{m} D^{-\ell_i} = -D^{-\ell_i} \ln D$$

proportional sein, d.h. es gibt ein $\lambda \in \mathbb{R}$, so daß

$$p_i = \lambda D^{-\ell_i} \ln D \quad \text{für alle } i \,.$$

Da sowohl die Summe aller p_i als auch die Summe der $D^{-\ell_i}$ gleich eins ist, muß $\lambda = 1/\ln D$ sein, d.h.

$$p_i = D^{-\ell_i} \quad \text{und} \quad \ell_i = -\log_D p_i \,.$$

Der Wert der Zielfunktion in diesem Punkt ist

$$L = \sum_{i=1}^{m} p_i \ell_i = -\sum_{i=1}^{m} p_i \log_D p_i \,,$$

in Falle $D = 2$ also die Entropie $H(X)$, ansonsten eine dazu proportionale Größe, die wir der Kürze halber mit $H_D(X)$ bezeichnen wollen.

Bislang haben wir nur eine *notwendige* Bedingung für ein Extremum gefunden; das folgende Lemma zeigt, daß wir tatsächlich das globale Minimum gefunden haben. (Der Leser sollte sich davon überzeugen, daß der Beweis auch funktioniert, wenn die ℓ_i beliebige reelle Zahlen sind, die der Ungleichung von KRAFT und MCMILLAN genügen.)

Lemma: Die mittlere Länge L eines jeden eindeutig dekodierbaren Quellencodes mit D-elementigem Zeichensatz ist mindestens gleich $H_D(X)$ mit Gleichheit genau dann, wenn für alle Wahrscheinlichkeiten p_i gilt: $p_i = D^{-\ell_i}$ mit einem $\ell_i \in \mathbb{N}_0$.

Beweis: Die Ungleichung von KRAFT und MCMILLAN sagt uns, daß $c = \sum_{i=1}^{m} D^{-\ell_i} \leq 1$ ist; setzen wir $q_i = D^{-\ell_i}/c$, so definieren auch die q_i eine Wahrscheinlichkeitsverteilung q und

$$\begin{aligned} L - H_D(X) &= \sum_{i=1}^{m} p_i \ell_i + \sum_{i=1}^{m} p_i \log_D p_i \\ &= -\sum_{i=1}^{m} p_i \log_D D^{-\ell_i} + \sum_{i=1}^{m} p_i \log_D p_i \\ &= \sum_{i=1}^{m} p_i \log_D \frac{p_i}{c r_i} = \sum_{i=1}^{m} p_i \log_D \frac{p_i}{r_i} - \sum_{i=1}^{m} p_i \log_D c \\ &= \frac{D(p\|q)}{\log_2 D} - \log_D c \geq 0 \,, \end{aligned}$$

da die KULLBACK-LEIBLER-Distanz zweier Wahrscheinlichkeitsverteilungen nicht negativ ist und $\log_D c$ wegen $c \leq 1$ nicht positiv sein kann. Falls die Differenz verschwindet, muß $c = 1$ und $D(p\|q) = 0$ sein, d.h. für jedes i ist $p_i = q_i = D^{-\ell_i}$.

■

Damit haben wir eine untere Grenze für L gefunden, die allerdings nur in recht speziellen Situationen wirklich angenommen wird. Wie der folgende Satz zeigt, können wir aber immer einen Code finden, für den sie um weniger als eins überschritten wird:

Satz: X sei eine Zufallsvariable mit Werten in einem m-elementigen Alphabet A, dessen i-tes Element mit Wahrscheinlichkeit p_i angenommen werde. Dann gibt es einen Quellencode für A mit einem D-elementigen Zeichensatz, dessen mittlere Länge L die Ungleichung

$$H_D(X) \leq L < H_D(X) + 1$$

erfüllt.

Beweis: Wir suchen natürliche Zahlen $\ell_1, \ldots, \ell_m$, für die $\sum p_i \ell_i$ möglichst klein wird, die aber die Ungleichung von KRAFT und MCMILLAN erfüllen. Im Reellen wird für $\ell_i = -\log_D p_i$ das Minimum angenommen; wir gehen zur nächstgrößeren ganzen Zahl, definieren ℓ_i also als diejenige natürliche Zahl, für die gilt

$$-\log_D p_i \leq \ell_i < -\log_D p_i + 1 \,.$$

Dann ist

$$\sum_{i=1}^{m} D^{-\ell_i} \leq \sum_{i=1}^{m} D^{\log_D p_i} = \sum_{i=1}^{m} p_i = 1 \,,$$

die Ungleichung von KRAFT und MCMILLAN ist somit erfüllt, so daß es einen Praefixcode C gibt, der dem i-ten Buchstaben des Alphabets A ein Codewort der Länge ℓ_i zuordnet. Für dessen mittlere Länge gilt

$$H_D(X) = \sum_{i=1}^{m} p_i(-\log_D p_i) \leq L = \sum_{i=1}^{m} p_i \ell_i$$

$$< \sum_{i=1}^{m} p_i(-\log_D p_i + 1) = H_D(X) + 1 \,.$$

In den ersten Paragraphen dieses Kapitels haben wir mehrere Beispiele betrachtet, bei denen sich die Annäherung der mittleren Bitzahl pro Buchstabe an die Entropie verbessern ließ, wenn wir statt einzelner Buchstaben Blöcke von Buchstaben betrachten. Der gerade bewiesene Satz erklärt auch das:

Korollar: X sei eine Zufallsvariable mit Werten in einem m-elementigen Alphabet A, dessen i-tes Element mit Wahrscheinlichkeit p_i angenommen werde. Dann gibt es einen Quellencode für A^n mit einem D-elementigen Zeichensatz, dessen mittlere Länge L pro Buchstaben aus A die Ungleichung

$$H_D(X) \leq L < H_D(X) + \frac{1}{n}$$

erfüllt.

Beweis: Wir betrachten ein n-tupel aus unabhängigen Zufallsvariablen $X_1, \ldots, X_n$, die allesamt Werte im Alphabet A annehmen und alle die gleiche Wahrscheinlichkeitsverteilung haben wie X. Die Entropie dieses n-tupels (zur Basis D) mit Werten in A^n ist $nH_D(X)$; nach dem gerade bewiesenen Satz gibt es also einen Quellencode, dessen mittlere Länge zwischen $nH_D(X)$ und $nH_D(X) + 1$ liegt. Die Länge L bezogen auf die Buchstaben aus A ist ein n-tel davon, erfüllt also die behauptete Ungleichung.

■

Dieses Korollar zeigt insbesondere, daß wir den mittleren Aufwand pro Buchstabe mit hinreichend großen Blocklängen beliebig nahe an die Entropie annähern können.

c) Huffman-Codes

DAVID HUFFMAN stellte 1951 ein Verfahren vor, wie man zu einer gegebenen Häufigkeitsverteilung einen Praefixcode mit minimaler mittlerer Länge konstruieren kann. In seiner Arbeit

> DAVID A. HUFFMAN: A Method for the Construction of Minimum-Redundancy Codes, *Proc. I.R.E., Sept. 1952, 1098–1101*

geht er aus von einer endlichen Menge von Nachrichten, also dem, was wir hier immer als das Alphabet $A = \{a_1, \ldots, a_m\}$ bezeichnen, und ordnet sie so, daß für die Wahrscheinlichkeit p_i von a_i gilt: $p_i \geq p_j$ falls $i < j$.

Für einen Code C mit minimaler mittlerer Länge kann man dann ohne Beschränkung der Allgemeinheit davon ausgehen, daß für die Länge ℓ_i

des Codeworts $C(a_i)$ gilt: $\ell_i \leq \ell_j$ fälls $i < j$. Wäre nämlich $\ell_j > \ell_i$, so wäre $p_j\ell_j + p_i\ell_i > p_i\ell_j + p_j\ell_i$, d.h. die mittlere Länge des Codes würde echt kleiner durch Vertauschen der Codewörter $C(a_i)$ und $C(a_j)$, im Widerspruch zur vorausgesetzten Minimalität.

Außerdem muß $\ell_m = \ell_{m-1}$ sein, denn die ersten ℓ_{m-1} Zeichen von $C(A_m)$ können wegen der Praefixbedingung kein Codewort für einen anderen Buchstaben sein; wenn wir etwaige folgende Zeichen von $C(a_m)$ streichen, erhalten wir einen neuen Praefixcode, dessen mittlere Länge im Fall $\ell_m > \ell_{m-1}$ kleiner wäre als die von C.

Somit gibt es mindestens zwei Buchstaben, denen ein Codewort maximaler Länge zugeordnet wird. Wir können aber noch mehr sagen: Es gibt unter den Codewörtern maximaler Länge mindestens zwei, die sich nur in ihrem letzten Zeichen unterscheiden: Wäre dies nicht der Fall, könnten wir bei allen Codewörtern maximaler Länge das letzte Zeichen streichen ohne die Praefixbedingung zu verletzen.

Bislang gilt alles für Codes mit einer beliebigen Anzahl D von Zeichen; für die Konstruktion eines Codes anhand der aufgestellten Prinzipien wollen wir uns aber – genau wie HUFFMAN – zunächst auf den Fall $D = 2$ beschränken, und die Modifikationen für $D > 2$ anschließend kurz diskutieren.

Angenommen, wir haben einen optimalen binären Code für ein Alphabet aus $m > 2$ Buchstaben. Dann wissen wir, daß es unter den Codewörtern maximaler Länge zwei gibt, die sich nur im letzten Bit unterscheiden; indem wir die Codes der Buchstaben mit maximaler Codelänge nötigenfalls permutieren, können wir annehmen, daß es sich dabei im die beiden Buchstaben a_m und a_{m-1} handelt. (Man beachte, daß HUFFMAN die Buchstaben nach ihrer Häufigkeit anordnet.)

Für einen beliebigen binären Code, bei dem die beiden Buchstaben geringster Wahrscheinlichkeit Codewörter maximaler Länge haben, die sich nur im letzten Bit unterscheiden, können wir die HUFFMAN-Reduktion bilden wie folgt:

Wir betrachten ein neues Alphabet A^* aus $m-1$ Buchstaben; es enthält die Buchstaben a_1 bis a_{m-2} sowie einen neuen Buchstaben a^*, der

mit Wahrscheinlichkeit $p_{m-1} + p_m$ auftreten soll. Zu diesem Alphabet betrachten wir den Code C^*, der den a_i mit $i \leq m-2$ das Codewort $C(a_i)$ zuordnet; $C^*(a^*)$ sei $C(a_m)$ ohne das letzte Bit. Umgekehrt läßt sich C aus C^* fast eindeutig rekonstruieren: Wir setzen $C(a_{m-1})$ auf $C^*(a^*)$ gefolgt von einer Null und $C(a_m)$ auf $C^*(a^*)$ gefolgt von einer Eins (oder umgekehrt). Die mittlere Länge L^* von C^* läßt sich leicht durch die mittlere Länge L von C ausdrücken:

$$L^* = \sum_{i=1}^{m-1} p_i \ell_i + (p_{m-1} + p_m)(\ell_m - 1) = \sum_{i=1}^{m} p_i \ell_i - p_{m-1} - p_m$$
$$= L - p_{m-1} - p_m \,,$$

denn $\ell_{m-1} = \ell_m$.

HUFFMANs Konstruktion beruht wesentlich auf der folgenden Beobachtung: *C ist genau dann optimal, wenn C^* optimal ist.*

Ist nämlich C^* nicht optimal, so gibt es einen Code D^* mit kleinerer mittlerer Länge $L^{**} < L^*$. Daraus läßt sich ein Code D für A konstruieren, bei dem $D(a_{m-1})$ und $D(a_m)$ aus $D^*(a^*)$ entstehen durch Anhängen einer Null *bzw.* einer Eins; die mittlere Länge dieses Codes ist $L^{**} + p_{m-1} + p_m < L$, so daß auch C nicht optimal ist. Entsprechend folgt auch die andere Richtung.

Damit ist die rekursive Struktur des Algorithmus von HUFFMAN klar: Wir wollen ein Alphabet A kodieren, das m Buchstaben enthält.

Im Fall $m = 2$ können wir einfach jedem der beiden Buchstaben eines der beiden Codezeichen zuordnen; die mittlere Länge des Codes ist dann eins, und kürzer kann sie bei keinem Code sein.

Ist $m > 2$, so führen wir eine HUFFMAN-Reduktion durch, d.h. wir fassen die beiden am wenigsten wahrscheinlichen Zeichen zusammen zu einem Zeichen a^*, dem wir die Summe von deren Wahrscheinlichkeiten zuordnen. Dank unserer rekursiven Vorgehensweise können wir für dieses Alphabet aus $m-1$ Elementen einen optimalen Code konstruieren; um auch einen für das gegebene Alphabet zu erhalten, kodieren wir die beiden seltensten Zeichen so, daß wir an das Codewort für a^*

eine Null *bzw.* eine Eins anhängen. Wie wir uns gerade überlegt haben, ist dann auch der neue Code optimal.

Als Beispiel betrachten wir eine Zufallsvariable X mit Werten in einem sechselementigen Alphabet $A = \{a, b, c, d, e, f\}$; die Wahrscheinlichkeiten der sechs Buchstaben seien (in alphabetischer Reihenfolge) $\frac{1}{3}, \frac{1}{4}, \frac{1}{6}, \frac{1}{8}, \frac{1}{12}$ und $\frac{1}{24}$.

Die beiden seltensten Zeichen sind e und f; wir fassen sie also zusammen zu einem Zeichen ef mit Wahrscheinlichkeit $\frac{1}{12} + \frac{1}{24} = \frac{1}{8}$. Im neuen Alphabet $\{a, b, c, d, ef\}$ sind d und ef die beiden seltensten Buchstaben; wir fassen sie also zusammen zu einem neuen Buchstaben $d(ef)$ mit Wahrscheinlichkeit $\frac{1}{8} + \frac{1}{8} = \frac{1}{4}$. Im Alphabet $\{a, b, c, d(ef)\}$ ist c der seltenste Buchstabe; für den zweitseltensten haben wir die Auswahl zwischen b und $d(ef)$, die jeweils mit Wahrscheinlichkeit $\frac{1}{4}$ auftreten und müssen uns für einen der beiden entscheiden. Um Klammern zu sparen fassen wir b und c zusammen zu einem neuen Buchstaben bc mit Wahrscheinlichkeit $\frac{1}{6} + \frac{1}{4} = \frac{5}{12}$. Dies führt auf das Alphabet $\{a, bc, d(ef)\}$, in dem $d(ef)$ und a die beiden seltensten Buchstaben sind; wir fassen sie zusammen zum „Buchstaben“ $a\big(d(ef)\big)$. Damit sind wir bei einem zweibuchstabigen Alphabet angelangt; einer der beiden optimalen Codes besteht darin, daß wir $a\big(d(ef)\big)$ mit Null und bc mit Eins kodieren.

Nun müssen wir nacheinander die HUFFMAN-Reduktionen rückgängig machen. Als erstes wird bc aufgespalten in b mit dem Code 10 und c mit dem Code 11; das Umgekehrte wäre natürlich genauso gut möglich. Sodann wir $a\big(d(ef)\big)$ aufgespalten in a und $d(ef)$ durch die Kodierung $a = 00$ und $d(ef) = 01$. Als nächstes wird $d(ef)$ aufgespalten durch $d = 010$ und $ef = 011$, und im letzten Schritt setzen wir $e = 0110$ und $f = 0111$.

Damit haben wir einen optimalen Code gefunden; seine mittlere Länge ist

$$\frac{1}{3} \cdot 2 + \frac{1}{4} \cdot 2 + \frac{1}{6} \cdot 2 + \frac{1}{8} \cdot 3 + \frac{1}{12} \cdot 4 + \frac{1}{24} \cdot 4 = 2\frac{3}{8} = 2,375\,,$$

also nur wenig größer als die Entropie

$$H = \frac{1}{3}\log_2 3 + \frac{1}{4}\log_2 4 + \frac{1}{6}\log_2 6 + \frac{1}{12}\log_2 12 + \frac{1}{24}\log_2 24$$
$$= \frac{4}{3} + \frac{5}{8}\log_2 3 \approx 2{,}324\,.$$

Falls wir optimale Codes mit einem Zeichensatz von $D > 2$ Elementen suchen, können wir im wesentlichen genauso vorgehen; allerdings können wir nun für jedes Alphabet mit höchstens D Elementen einen Code mit mittlerer Länge eins finden. Bei jeder HUFFMAN-Reduktion außer der ersten fassen wir die D am wenigsten wahrscheinlichen Buchstabe zu einem neuen Buchstaben a^* zusammen; dadurch verringert sich die Elementanzahl dies Alphabets um $D-1$. Falls $m-1$ durch $D-1$ teilbar ist, können wir dies auch im ersten Schritt tun; andernfalls fassen wir dort nur m_0 Elemente zusammen, wobei $2 \le m_0 < D$ so gewählt wird, daß $D - m_0$ durch $D-1$ teilbar ist.

DAVID ALBERT HUFFMAN (1925–1999) studierte Elektrotechnik an der Ohio State University; nachdem er 1944 im Alter von 18 Jahren seinen Bachelor-Abschluß bekommen hatte, verbrachte er den Rest der Kriegszeit als Radaroffizier auf einem Schiff der US Navy. Nach dem Krieg kehrte er an die Ohio State University zurück; nach seinem Master 1949 wechselte er zur Promotion ans MIT. wo er danach von 1953 bis 1957 auch lehrte. Den HUFFMAN-Code entwickelte er dort als Studienarbeit während seines Promotionsstudiums. 1957 gründete er das Computer Science Department der Universität von Kalifornien in Santa Cruz, an der er 1994 emeritiert wurde. Die meisten seiner Arbeiten beschäftigen sich mit Informations- und Kodierungstheorie.

d) Komprimierung durch Dekorrelation

HUFFMAN-Codes sind optimal, wenn wir eine einzige Zufallsvariable betrachten oder, was auf dasselbe hinausläuft, eine Folge von unabhängigen identisch verteilten Zufallsvariablen. Eines der Haupteinsatzgebiete der Datenkompression sind aber Bild- und Audiodaten, bei denen diese Annahme sicherlich nicht erfüllt ist: Bei einer digitalen Tonaufnahme etwa wird der Schalldruck 44 100-mal pro Sekunde gemessen und auf einen Wert zwischen 0 und $2^{24} - 1 = 16\,777\,215$ oder 0 und $2^{16} - 1 = 65\,535$ skaliert. Eine Aufnahme, bei der die Lautstärke 44 100-mal pro Sekunde zufällig wechselt, werden nur wenige Hörer als Lieblingsmusik wählen: Dramatische Wechsel in der Lautstärke sind

zwar ein wichtiges kompositorisches Element, aber es muß sparsam eingesetzt werden. Von den 44 100 Werten, die pro Sekunde aufgezeichnet werden, unterscheidet sich die überwiegende Mehrzahl nur wenig von ihrem Vorgänger.

Ähnlich ist es bei Bilddaten: Selbstverständlich sind abrupte Übergänge auch hier ein oft eingesetztes Stilmittel, aber verglichen mit der Anzahl der Pixel, mit denen Bilder typischerweise digitalisiert werden, unterscheidet sich auch hier der Großteil aller Farb- oder Grauwerte nur wenig von den entsprechenden Werten der Nachbarpixel. Hier werden Grau- oder Farbwerte typischerweise nur mit Werten zwischen 0 und $2^8 - 1 = 256$ kodiert, da unser Auge bei gedruckten oder auf eine Leinwand projizierten Bildern selbst bei nur 64 verschiedenen Werten keine Artefakte mehr erkennen kann, wohingegen unser Gehör noch auf sehr viel feinere Unterschiede reagiert.

In beiden Fällen steckt also ein zumindest im Mittel wesentlicher Teil der Information bereits im Vorgänger; wir können dies dadurch quantifizieren, daß wir die typische Korrelation eines Schalldruck oder Farbwerts mit seinem Vorgänger (oder sonstigen Nachbar) berechnen. Diese Korrelation bezeichnet man als die *Autokorrelation* des stochastischen Prozesses, durch den wir ein Musikstück oder Bild beschreiben.

Auf der folgenden Doppelseite sind einige in Lehrbüchern der Bildverarbeitung beliebte Grauwerttestbilder zu sehen zusammen mit den Daten über minimale, maximale und mittlere Helligkeit $x_{\min}, x_{\max}$ und μ, Varianz σ^2, Standardabweichung σ sowie der Autokorrelation ρ. Diese aus

> P.M. FARELLE: Recursive Block Coding for Image Data Compression, *Springer,* 1990

entnommenen Daten beziehen sich natürlich auf die Originalbilder und nicht auf das, was Ihr Bildschirm oder Drucker daraus macht. Trotzdem sollte der Vergleich von Bildern und Daten einen einigermaßen korrekten Eindruck zumindest der relativen Situation vermitteln, da hoffentlich alle hier abgedruckten Bilder in derselben Weise verunstaltet sind.

Wie die Daten zeigen, können wir bei der Komprimierung von Bilddaten zumindest ungefähr von einer Autokorrelation um die 95% ausgehen;

jeder Wert ist somit im Mittel bereits zu 95% durch seinen Vorgänger bestimmt. Trotzdem übertragen oder speichern wir zumindest mit den uns bislang bekannten Kompressionsverfahren immer wieder 100% der Information einer jeden Zufallsvariablen.

Ein möglicher Ansatz zur Datenkompression wäre daher, daß wir immer nur die Differenz zum Vorgänger übertragen oder speichern: Haben X und Y beide Erwartungswert μ und Varianz σ^2, so hat $Z = Y - X$ den deutlich kleineren Erwartungswert $(1 - \rho)\mu$ und nur noch Varianz $2(1 - \rho)\sigma^2$. Die Kovarianz zwischen X und Z ist

$$\begin{aligned}\mathrm{Cov}(X, Z) &= \mathrm{Cov}(X, Y - \rho X) = \mathrm{Cov}(X, Y) - \rho\,\mathrm{Cov}(X, X)\\ &= \rho\sigma^2 - \rho\sigma^2 = 0\,;\end{aligned}$$

X und Z sind also unabhängig voneinander.

Verfahren, die dies ausnutzen, gibt es in der Tat; sie müssen aber mit dem Problem fertig werden, daß die Differenz zwar *meistens* klein ist, daß aber gerade die Ausnahmen, bei denen sie groß ist, sehr wesentlich für die Rekonstruktion des Original sind.

Der am häufigsten verwendete Ansatz geht daher anders vor: Die „Nachricht“ wird in Blöcke aufgeteilt; bei der Schallaufzeichnung auf CD sind dies bei den derzeit auf dem Massenmarkt erhältlichen Geräten Blöcke zu je acht Werten, bei Bildern sind es Teilbilder von jeweils 8×8 Pixel.

Anstelle der einzelnen Zufallsvariablen kodieren wir die Blöcke; wie wir bereits mehrfach gesehen haben, läßt sich allein dadurch der mittlere Aufwand meistens reduzieren, wenn auch nicht so dramatisch wie bei den hier betrachteten Komprimierungsverfahren.

Der Einfachheit halber beschränken wir uns auf ein eindimensionales Modell, mit dem wir es beispielsweise bei Audiodaten zu tun haben. Wir betrachten Blöcke aus einer gewissen Anzahl n von Zufallsvariablen, die allesamt dasselbe in $\mathbb{R}$ enthaltene Alphabet und dieselbe Wahrscheinlichkeitsverteilung haben. Dabei nehmen wir an, daß jede der Zufallsvariable mit ihrem Nachbarn die Korrelation ρ habe.

Als erstes sollten wir uns fragen, wie es mit der Korrelation zwischen weiter entfernten Variablen aussieht. Dazu können wir *a priori* nichts

Peppers

μ = 115,6
σ^2 = 5632
σ = 75,0
ρ = 0,98
$x_{\min}$ = 0
$x_{\max}$ = 237

Lenna

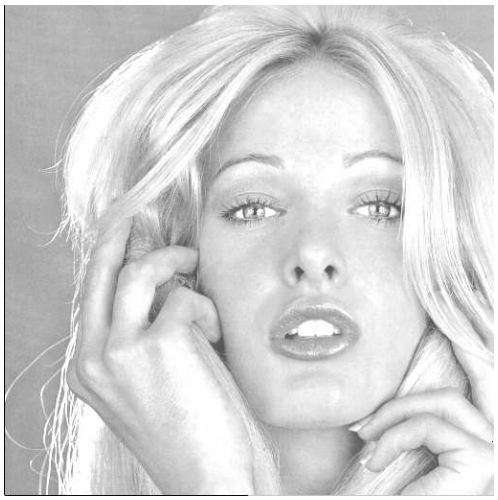
μ = 99,1
σ^2 = 2796
σ = 52,9
ρ = 0,97
$x_{\min}$ = 3
$x_{\max}$ = 248

Sailboat

μ = 124,3
σ^2 = 6027
σ = 77,6
ρ = 0,97
$x_{\min}$ = 0
$x_{\max}$ = 249

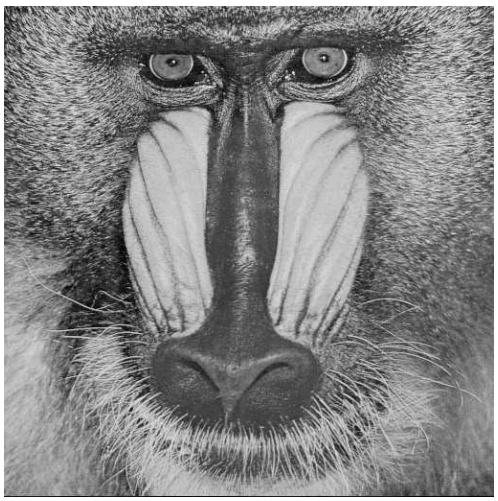
Stream

μ = 113,8
σ^2 = 2996
σ = 54,7
ρ = 0,94
$x_{\min}$ = 0
$x_{\max}$ = 255



Tiffany

μ = 208,6
σ^2 = 1126
σ = 33,6
ρ = 0,87
$x_{\min}$ = 3
$x_{\max}$ = 255



Baboon

μ = 128,9
σ^2 = 2282
σ = 47,8
ρ = 0,86
$x_{\min}$ = 0
$x_{\max}$ = 236

sagen; wenn wir im Extremfall nur zwei Zufallsvariablen mit Korrelation ρ haben, so daß alle Folgenglieder mit geradem Index gleich der einen und alle übrigen gleich der anderen sind, ist die Korrelation gleich ρ, wenn sich die Indizes um eine ungerade Zahl unterscheiden, und eins sonst. Dieser Fall wird freilich bei Bild- und Audiodaten kaum vorkommen.

Dort geht man üblicherweise aus vom sogenannten ar(1)-Modell, wonach – in Analogie zu MARKOV-Ketten – alle Abhängigkeiten ausschließlich auf die zwischen unmittelbaren Nachbarn zurückzuführen sind, so daß die Korrelation zwischen zwei Zufallsvariablen gleich ρ hoch Betrag der Indexdifferenz ist.

Beim blockweisen Kodieren nach diesem Modell betrachten wir somit einen Vektor $(X_1, \ldots, X_n)$ von Zufallsvariablen; alle X_i haben denselben Erwartungswert μ und dieselbe Varianz σ^2; die Korrelation zwischen X_i und X_j ist $\rho^{|i-j|}$. Die Korrelationsmatrix, bei der an der Stelle ij dieser Wert steht, ist somit die symmetrische Matrix

$$\begin{pmatrix} 1 & \rho & \rho^2 & \ldots & \rho^{n-1} \\ \rho & 1 & \rho & \ldots & \rho^{n-2} \\ \rho^2 & \rho & 1 & \ldots & \rho^{n-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \rho^{n-1} & \rho^{n-2} & \rho^{n-3} & \ldots & 1 \end{pmatrix};$$

die Kovarianzmatrix ist das σ^2-fache davon.

Wir betrachten nun anstelle der Zufallsvariablen X_i neue Variablen

$$Y_i = \sum_{j=1}^{n} a_{ij} X_j$$

mit zunächst irgendwelchen reellen Zahlen a_{ij}. Dann ist

$$\begin{aligned} \operatorname{Cov}(Y_i, Y_k) &= \operatorname{Cov}\left(\sum_{j=1}^{n} a_{ij} X_j, \sum_{\ell=1}^{n} a_{k\ell} X_\ell \right) \\ &= \sum_{j=1}^{n} \sum_{\ell=1}^{n} a_{ij} \operatorname{Cov}(X_j, X_\ell) a_{k\ell} \,. \end{aligned}$$

Stünde in der letzten Summe ganz rechts $a_{\ell k}$ anstelle von $a_{k\ell}$, so wäre die Summe gerade der ik-Eintrag des Produkts A mal Kovarianzmatrix der X_i mal A, wobei A die Matrix mit Einträgen a_{ij} bezeichnet. Da tatsächlich $a_{k\ell}$ dasteht, müssen wir als letzten Faktor des Matrixprodukts die transponierte Matrix A^T anstelle von A nehmen; die Kovarianzmatrix der neuen Zufallsvariablen ist also

$$\operatorname{Cov}(Y_1, \ldots, Y_n) = A \operatorname{Cov}(X_1, \ldots, X_n) A^T \, .$$

Als nächstes beachten wir, daß die Kovarianzmatrix der X_i symmetrisch ist; nach dem (im Anhang zu diesem Paragraphen bewiesenen) Spektralsatz gibt es daher eine Orthonormalbasis des $\mathbb{R}^n$, bezüglich derer sie Diagonalgestalt hat. Es gibt also eine Matrix A, so daß $A \operatorname{Cov}(X_1, \ldots, X_n) A^{-1}$ eine Diagonalmatrix ist, und da die Matrix A zu einem Basiswechsel zwischen zwei Orthonormalbasen gehört, ist AA^T die Einheitsmatrix, d.h. $A^{-1} = A^T$.

Definieren wir daher $Y_i = \sum a_{ij} X_j$ mit den Einträgen dieser Matrix A, so ist $\operatorname{Cov}(Y_1, \ldots, Y_n)$ eine Diagonalmatrix; die verschiedenen Y_i sind also voneinander unabhängige Zufallsvariablen.

Unter den Annahmen des ar(1)-Modells können wir somit jede Folge von Zufallsvariablen durch eine lineare Transformation in eine Folge unkorrelierter Zufallsvariablen überführen. Diese Transformation bezeichnet man, obwohl sie zuerst von HOTELLING vorgeschlagen wurde, als KARHUNEN-LOÈVE-Transformation.

HAROLD HOTELLING (1895–1973) war ein amerikanischer Statistiker und Ökonom; er lehrte an der Columbia University und der University of North Carolina. In einer 1933 veröffentlichten Arbeit im *Journal of Educational Psychology* schlug er erstmalig diese Transformation vor, die von Statistikern heute in Anlehnung an den Titel seiner Arbeit meist als *Hauptkomponentenanalyse* bezeichnet wird. In Europa erschien die Transformation fast gleichzeitig um 1947 *bzw.* 1948 in wahrscheinlichkeitstheoretischen Arbeiten des Finnen KARI KARHUNEN (1915–1992) und des Franzosen MICHEL LOÈVE (1907–1979), nach denen sie in der technischen Literatur benannt wird.

Die Matrix A der linearen Transformation hängt nur von ρ ab und kann daher für gängige Werte von ρ vorberechnet werden; die KARHUNEN-LOÈVE-Transformation besteht somit einfach in der Multiplikation mit einer bekannten Matrix. Da die Abhängigkeit von ρ im hier relevanten Bereich relativ schwach ist, verschlechtern sich die Ergebnisse kaum, wenn man sich dabei auf *ein* typisches ρ beschränkt, etwa auf $\rho = 0{,}95$.

Trotzdem wird die KARHUNEN-LOÈVE-Transformation praktisch nie verwendet, denn durch einen anderen Ansatz kommt man mit deutlich geringerem Aufwand auf fast dasselbe Ergebnis. Um zu sehen, wie dieser Funtkioniert, betrachten wir die für Komprimierungsverfahren in der Unterhaltungselektronik typischen Werte $n = 8$ und $\rho = 0{,}95$. Die Eigenvektoren der Kovarianzmatrix

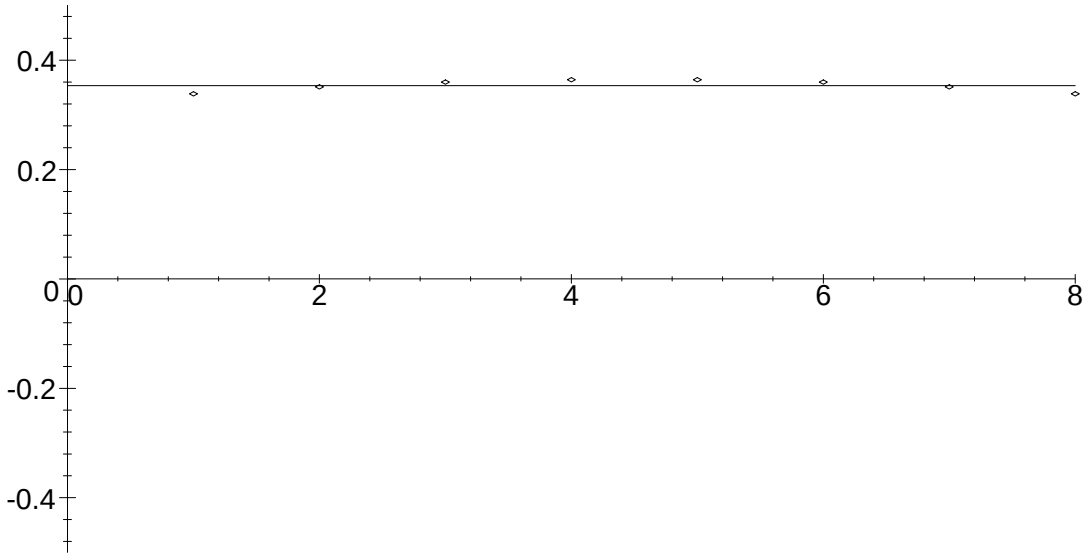
$$\mathrm{Cov}(X_1, \ldots, X_8) = \begin{pmatrix} 1 & \rho & \rho^2 & \rho^3 & \rho^4 & \rho^5 & \rho^6 & \rho^7 \\ \rho & 1 & \rho & \rho^2 & \rho^3 & \rho^4 & \rho^5 & \rho^6 \\ \rho^2 & \rho & 1 & \rho & \rho^2 & \rho^3 & \rho^4 & \rho^5 \\ \rho^3 & \rho^2 & \rho & 1 & \rho & \rho^2 & \rho^3 & \rho^4 \\ \rho^4 & \rho^3 & \rho^2 & \rho & 1 & \rho & \rho^2 & \rho^3 \\ \rho^5 & \rho^4 & \rho^3 & \rho^2 & \rho & 1 & \rho & \rho^2 \\ \rho^6 & \rho^5 & \rho^4 & \rho^3 & \rho^2 & \rho & 1 & \rho \\ \rho^7 & \rho^6 & \rho^5 & \rho^4 & \rho^3 & \rho^2 & \rho & 1 \end{pmatrix}$$

können dann zumindest numerisch leicht bestimmt werden; da man acht Vektoren mit jeweils acht Einträgen wenig Struktur ansehen kann, habe ich die Ergebnisse in den folgenden acht Zeichnungen graphisch dargestellt: Die eingezeichneten Punkte sind (i, x_i) für $i = 1, \ldots, 8$, wobei x_i jeweils die i-te Komponente des auf Länge eins normierten Eigenvektors bezeichnet. Zusätzlich ist in der j-ten Zeichnung noch die Kurve

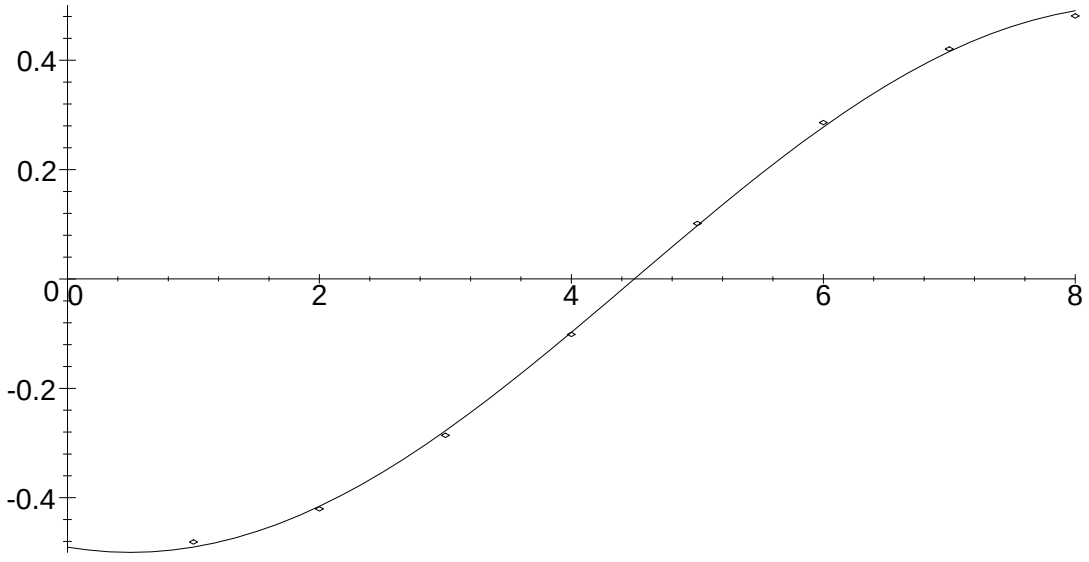
$$y = \cos\left(\frac{(2x-1)(j-1)\pi}{16}\right)$$

eingezeichnet; wie man sieht, liegen die Punkte (i, x_i) zwar nicht exakt auf diesen Kurven, aber doch sehr in deren Nähe. Entsprechendes gilt auch für andere Werte von n und andere Korrelationskoeffizienten ρ nahe eins.

Der Grund dafür, daß man in der Praxis lieber mit den so durch Kosinuswerte angenäherten Basisvektoren arbeitet, liegt nicht darin, daß

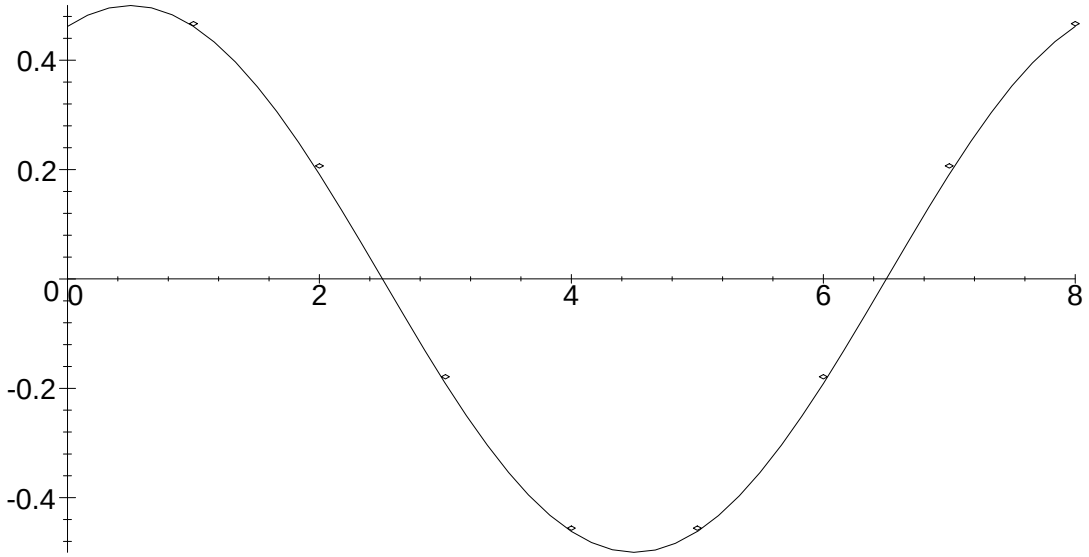


Der erste Eigenvektor der Korrelationsmatrix

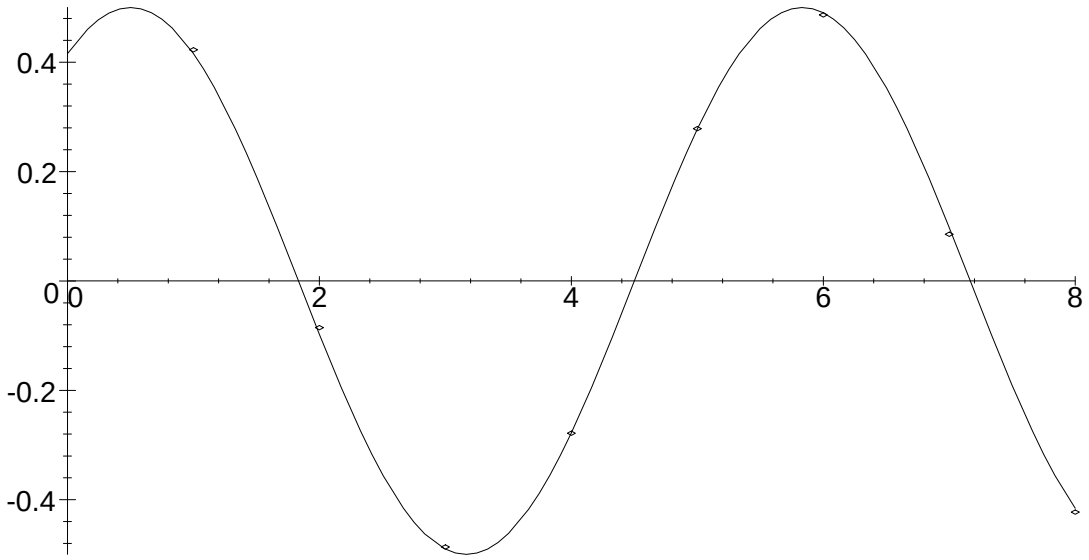


Der zweite Eigenvektor der Korrelationsmatrix

diese einfacher zu berechnen sind – die Eigenvektoren sind schließlich Konstanten des Komprimierungsverfahren. Für die Transformation eines Blocks in die neue Basis brauchen wir aber eine Matrix-Vektor-

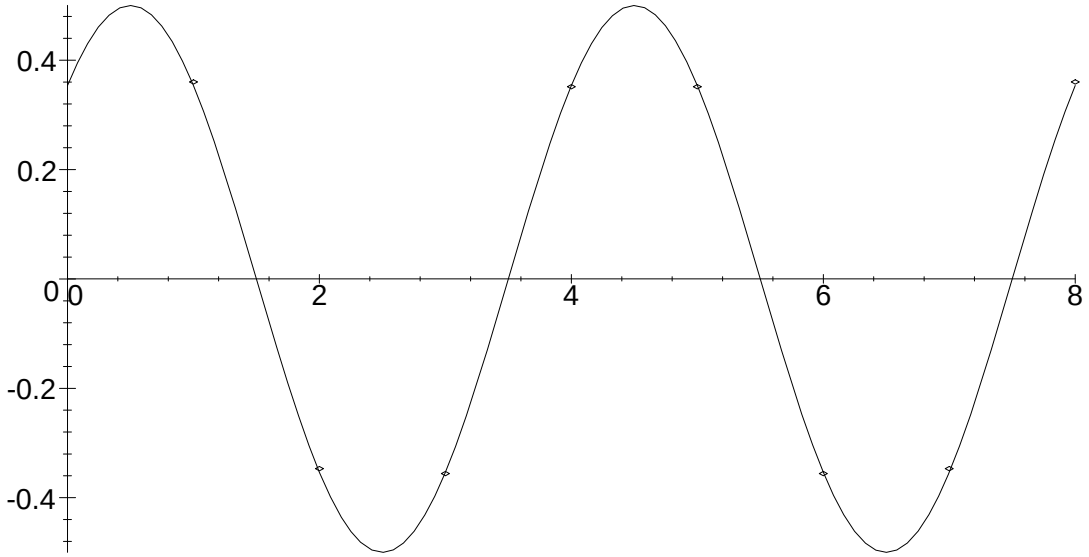


Der dritte Eigenvektor der Korrelationsmatrix

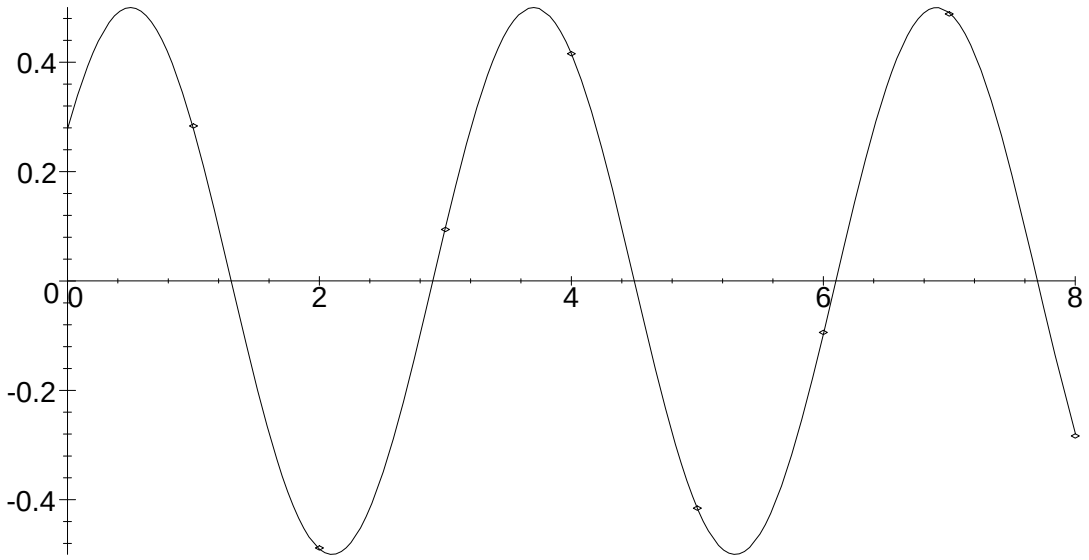


Der vierte Eigenvektor der Korrelationsmatrix

Multiplikation, d.h. n^2 Multiplikationen sowie $n(n-1)$ Additionen reeller Zahlen. Wenn wir stattdessen mit den durch Kosinuswerte gegebenen Vektoren arbeiten, können wir eine sogenannte schnelle FOURIER-

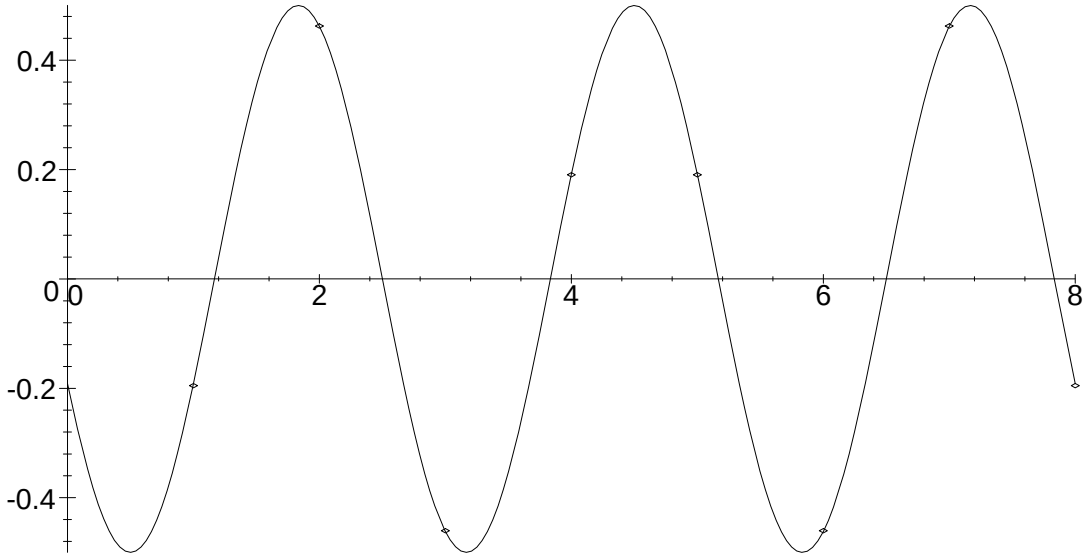


Der fünfte Eigenvektor der Korrelationsmatrix

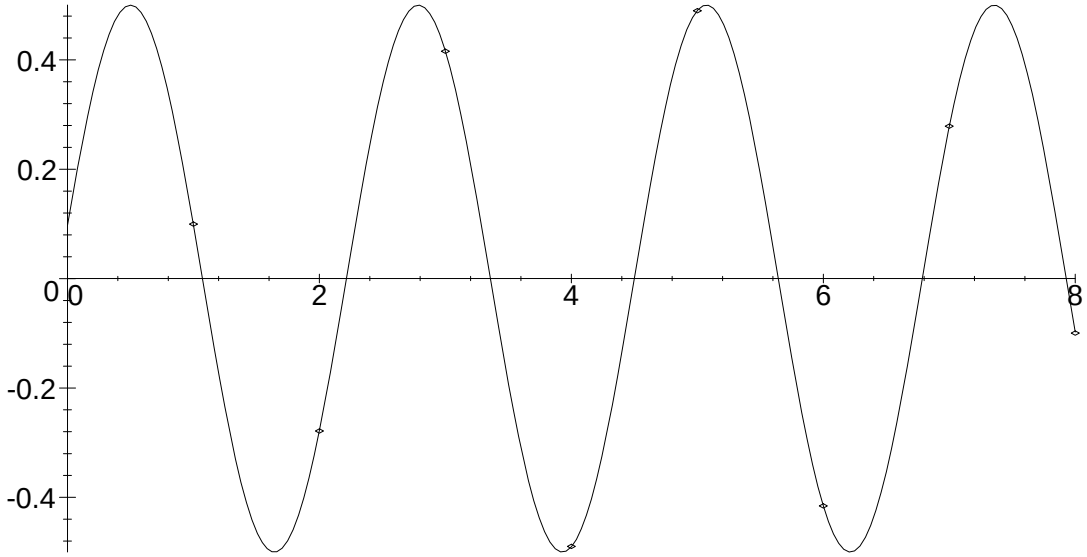


Der sechste Eigenvektor der Korrelationsmatrix

bzw. Kosinustransformation anwenden, die nur ungefähr $n \log_2 n$ Multiplikationen benötigt. Ihre Dekorrelationseffiziens liegt im Fall $n = 8$ und $\rho \approx 0{,}95$ bei etwa 98%; wir verlieren also fast nichts im Vergleich



Der siebte Eigenvektor der Korrelationsmatrix



Der achte Eigenvektor der Korrelationsmatrix

zur aufwendigeren KARHUNEN-LOÈVE-Transformation.

Schnelle FOURIER- und Kosinustransformationen gibt es auch in höheren Dimensionen; insbesondere können wir im Zweidimensionalen Blöcke

von 8×8 Pixel schnell so in eine neue Basis des 64-dimensionalen Raums transformieren, daß sie praktisch unabhängig voneinander sind.

Die diskrete Kosinustransformation ist Teil fast aller gängiger Normen zur Bildkomprimierung: Sowohl der JPEG-Standard für Photographien, die Standards MPEG 1 und 2 für digitale (Unterhaltungs-)Videos als auch der Standard CCITT H.261 für Videokonferenzen enthalten (neben anderen Bestandteilen) jeweils eine diskrete Kosinustransformation; auch im mp3-Standard ist sie ein Teil der Kodierung.

Die Transformation allein ist natürlich noch keine Komprimierung: Schließlich haben wir nur einen Vektor in einer anderen Basis hingeschrieben, und die Anzahl der reellen Zahlen, die man zur Beschreibung eines solchen Vektors benötigt, ist unabhängig von der Basis. Der wesentliche Vorteil der neuen Basis ist, daß man statistisch recht gute Aussagen über die Größe der Komponenten machen kann. Hier wollen wir auf exakte statistische Berechnungen verzichten und stattdessen informell diskutieren, warum dies der Fall sein könnte.

Wie die Abbildungen der Basisvektoren zur KARHUNEN-LOÈVE-Transformation und die Formeln für die Basisvektoren zur diskreten Kosinustransformation zeigen, werden die Basisvektoren, wenn man sie in der hier angegebenen Reihenfolge betrachtet, immer hochfrequenter. Von einem hinreichend fein abgetasteten Bild- oder Audiosignal erwarten wir, daß hochfrequente Schwankungen keine große Rolle spielen und somit die entsprechenden Basisvektoren nur kleine Koeffizienten haben oder in vielen Fällen sogar gleich gar nicht auftreten. Dementsprechend genügt es, für die Übertragung dieser Koeffizienten nur wenige Bits bereitzustellen; bei nur geringen Abstrichen an die Qualität kann man auf gewisse Koeffizienten sogar ganz verzichten.

Ein Kompressionsverfahren wird daher, je nach Anspruch an die Qualität, entweder alle Koeffizienten des Signals in der neuen Basis übertragen und durch eine geeignete Darstellung der Daten dafür sorgen, daß Folgen von Nullen nur wenig Platz benötigen, oder aber es wird nur eine Auswahl der Koeffizienten übertragen und auch für diese jeweils festlegen, wie viele Bit dafür in Anspruch genommen werden. Diese Anzahl wird umso geringer sein, je höher die Frequenz des jewei-

ligen Basisvektors ist; bei einigen Verfahren wie etwa JPEG können die Anzahlen auch variabel in Abhängigkeit von einer Qualitätszahl gewählt werden.

Zum Schluß sei noch ganz kurz erwähnt, daß die KARHUNEN-LOÈVE-Transformation und damit (mit ganz geringen Abstrichen) auch die diskrete Kosinustransformation zwar die Korrelationsmatrix in optimaler Weise diagonalisieren, daß aber daraus nicht folgt, daß sie auch optimale Kompressionsverfahren liefern: Ausßer der Kovarianz gibt es noch weitere Quellen für Redundanz eines Bildes.

Ein gewisser Nachteil der Kosinustransformation ist außerdem, daß man für abrupte Übergänge, wie sie etwa bei Kanten immer wieder einmal auftauchen, die hochfrequenten Basisvektoren braucht, die dann aber nicht nur die Kante selbst beeinflussen, sondern das gesamte Quadrat, auf das die Transformation angewandt wird.

Eine bessere Möglichkeit wäre es daher, wenn man anstelle von Kosinusfunktionen Funktionen verwenden könnte, die sowohl im Zeit- als auch im Frequenzbereich lokalisiert sind. Solche Funktionen gibt es in der Tat, etwa die sogenannten *Wavelets.* Hierbei handelt es sich um schnell abklingende Wellen, und neuere Arbeiten deuten darauf hin, daß diese für gewisse Bildmodelle (die im Gegensatz zum hier betrachteten nicht mit Wahrscheinlichkeiten arbeiten) nicht zu weit vom Optimum entfernt sein sollten. Im Rahmen dieser Vorlesung ist es jedoch zeitlich weder möglich, auf diese Modelle einzugehen, noch ist an eine genauere Behandlung von Wavelets zu denken.

Einen allgemein verständlichen Überblick über Wavelets findet man etwa bei

BARBARA BURKE HUBBARD: Wavelets: Die Mathematik der kleinen Wellen, *Birkhäuser* 1997;

das zitierte Optimalitätsresultat ist beschrieben im Vortrag

STÉPHANE MALLAT: Applied Mathematics meets signal processing

auf dem Internationalen Mathematikerkongress 1998 in Berlin, nachzulesen in Band I der Proceedings, S. 319–338, oder unter http://www.mathematik.uni-bielefeld.de/documenta/xvol-icm/00/Mallat.MAN.html.

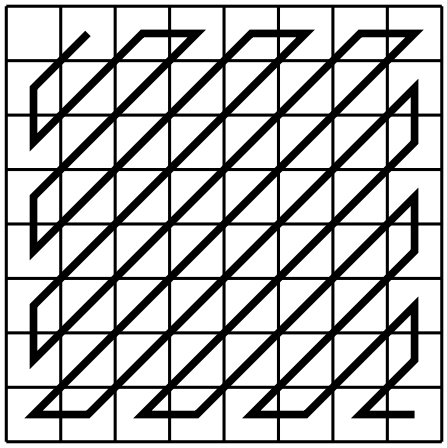
e) Datenkomprimierung bei JPEG

Als praktisches Beispiel eines Komprimierungsalgorithmus wollen wir zumindest kurz den des JPEG-Standards der *Joint Photographers Expert Group* betrachten, der in vielen Digitalkameras verwendet wird.

Er beginnt damit, das Bild in Blöcke von 8×8 Pixel aufzuteilen und jeden wie im vorigen Abschnitt beschrieben einer Kosinustransformation zu unterziehen. Das Ergebnis ist eine neue 8×8-Matrix reeller Zahlen, die zunächst quantisiert, d.h. auf Werte aus einer diskreten Menge gerundet werden. Größe und Aufbau dieser Menge hängen sowohl von der Position in der Matrix als von einem wählbaren Qualitätsfaktor ab.

Der Matrixeintrag links oben entspricht dem Basisvektor, dessen sämtliche Einträge gleich sind; dort steht also der Mittelwert der Einträge der ursprünglichen 8×8-Matrix. Er wird (abgesehen natürlich vom ersten Block) gespeichert als die Differenz vom Mittelwert des Vorgängerblocks; Vorgänger bezieht sich dabei auf die zeilenweise Anordnung.

Die übrigen 63 Einträge eines jeden Blocks werden mäanderförmig nach dem Schema im nächsten Bild durchlaufen:



Die so erhaltene Folge von 63 Werten wird meist viele Nullen enthalten; gespeichert werden daher nur die von Null verschiedenen Werte zusammen mit der Anzahl der Nullfelder vor so einem Wert.

Im letzten Schritt schließlich wird die gesamte so erhaltene Zeichenfolge HUFFMAN-kodiert.

Kapitel 2
Der wirtschaftliche Wert von Information

In diesem Kapitel wollen wir sehen, wie wir den Wert der Information einer Zufallsvariablen quantifizieren können. Dieser Wert hängt natürlich stark vom Umfeld ab und kann nur in Abhängigkeit davon betrachtet werden.

§1: Kellys Ansatz für Wetten

Siebeneinhalb Jahre, nachdem SHANNONs Arbeit *A Mathematical Theory of Communication* erschienen war, veröffentlichte J.L. KELLY JR., ein anderer Mitarbeiter der (im Rahmen der Öffnung des amerikanischen Telephonmarkts inzwischen aufgeteilten) *Bell Telephone Laboratories,* im Juli 1956 eine (heute auch im Internet zu findende) Arbeit mit dem Titel *A New Interpretation of Information Rate,* in der er einen neuen Zugang zu SHANNONs Ergebnissen vorstellte.

JOHN LARRY KELLY JR. (1923–1965) war während des zweiten Weltkriegs vier Jahre lang Pilot bei der US Air Force; nach dem Krieg begann er ein Physikstudium an der University of Texas in Austin, das er 1953 mit der Promotion abschloß. Nach dem Studium arbeitete er bei den Bell Labs unter anderem auf dem Gebiet der Sprachsynthese. Bei der Anwendung der Informationstheorie auf die Spieltheorie arbeitete er eng mit SHANNON zusammen, der die Resultate im Gegensatz zu KELLY auch wirklich in Las Vegas anwendete.

SHANNON hatte hatte seine Theorie insbesondere angewandt auf einen gestörten Kommunikationskanal. Darunter versteht man im einfachsten Fall eine Leitung oder sonstige Verbindung, über die Bits übertragen

werden sollen. Da dies in der Praxis nie ganz ohne Störungen möglich ist, gibt es eine gewisse Fehlerwahrscheinlichkeit p; beim sogenannten symmetrischen Kanal ist diese unabhängig vom übertragenen Symbol, d.h. mit Wahrscheinlichkeit p wird eine Null zur Eins und umgekehrt.

Mathematisch können wir dies folgendermaßen formulieren: Wir betrachten die zu übertragenden Nachrichten als eine Bitfolge $x_1, x_2, \ldots$ und betrachten eine zusätzliche Quelle, die diese Übetragung stört. Diese Quelle produziert ebenfalls eine Bitfolge $y_1, y_2, \ldots$, und der Empfänger erhält an Stelle der ursprünglichen Nachricht eine Nachricht $z_1, z_2, \ldots$ mit

$$z_i = \begin{cases} x_i & \text{falls } y_i = 0 \\ 1 - x_i & \text{falls } y_i = 1 \end{cases} .$$

Das i-te Bit wird somit genau dann verändert, wenn $y_i = 1$ ist; ansonsten wird es korrekt übertragen. (Einfacher könnte man obige Formel auch schreiben als $z_i = x_i + y_i \bmod 2$.)

Da Fehler mit Wahrscheinlichkeit p auftreten, produziert die Fehlerquelle Einsen mit Wahrscheinlichkeit p und Nullen mit Wahrscheinlichkeit $q = 1 - p$; ihre Entropie ist also $-p \log p - q \log q$.

Diese Information fehlt dem Empfänger: Wenn wir davon ausgehen, daß jedes gesendete Bit eine Information von einem Bit enthält, ist die Information pro empfangenes Bit nur noch

$$r = 1 + p \log p + q \log q \text{ Bit} .$$

Diese Zahl r bezeichnet SHANNON als die Informationsrate des Kanals; er zeigt, daß man durch geeignete fehlerkorrigierende Codes die Nachricht so verschlüsseln kann, daß man dieser Rate beliebig nahe kommt: Zumindest im Prinzip kann der Empfänger also seine Nachricht so kodieren, daß ihre Länge nur um einen Faktor $1/r$ steigt und dann vom Empfänger mit einer Wahrscheinlichkeit beliebig nahe bei eins rekonstruiert werden kann.

KELLY wollte diese Informationsrate unabhängig von Kodierung definieren und ging dazu von folgender Situation aus: Er betrachtet die gesendeten Symbole als Ergebnis eines Zufallsprozesses, auf dessen

Ausgang gewettet werden kann. Der Wetter erhält auf diese Weise Informationen über den Ausgang des Ereignisses *bevor* diese allgemein bekannt wird und kann daher noch Wetten abschließen unter Berücksichtigung der erhaltenen Information.

Falls die Übertragung stets störungsfrei verliefe, könnte er bedenkenlos sein ganzes Vermögen V_0 einsetzen, denn er würde ja auf ein Ereignis wetten, von dem er mit Sicherheit weiß, daß es eintrat. Bei einer fairen Wette, bei der *a priori* beide Ausgänge gleich wahrscheinlich waren, würde sein Einsatz verdoppelt. und falls er jedes übertragene Bit entsprechend Nutzen kann, und jeweils das gesamte vorhandene Kapital einsetzt, wäre nach n Bit sein Vermögen angestiegen auf $V_n = V_0 \cdot 2^n$, würde also exponentiell wachsen.

Wenn der Übertragungskanal allerdings mit Wahrscheinlichkeit p das falsche Ergebnis überträgt, hat er für $p < \frac{1}{2}$ zwar immer noch einen Informationsvorsprung, muß aber damit rechnen, daß sein Einsatz mit Wahrscheinlichkeit p verloren geht. Der Erwartungswert für den Multiplikator bei einmaligem Wetten wäre $2q$, nach n-maligem Wetten also $(2q)^n$, was für $p < \frac{1}{2}$ und damit $q > \frac{1}{2}$ immer noch exponentiell ansteigt.

Dieser Erwartungswert ist ein gewichtetes Mittel aus nur zwei Zahlen: Falls alle n Symbole korrekt empfangen wurden, was mit Wahrscheinlichkeit q^n passiert, wird der ursprüngliche Einsatz mit 2^n multipliziert; andernfalls wurde mindestens einmal auf den falschen Ausgang gewettet, und das Kapital ist weg. Bei den nachfolgenden Wetten werden höchstens noch Nullen verdoppelt.

Die Wahrscheinlichkeit für einen Totalverlust beträgt also $1 - q^n$, was wegen $q < 1$ für hinreichend große Werte von n der Eins beliebig nahe kommt. In dieser Situation werden daher nur sehr harte Zocker ihr gesamtes Kapital einsetzen; KELLY betrachtet stattdessen einen Wetter, der bei jeder Wette nur einen gewissen Teil ℓ des vorhandenen Kapitals einsetzt. Falls er gewinnt, wird dieses Teil verdoppelt; ein Kapital K wird also zu

$$2 \cdot \ell K + (1 - \ell)K = (1 + \ell)K \, .$$

Im Verlustfall ist der Anteil ℓ verloren; das neue Kapital ist also nur noch $(1 - \ell)K$.

Bei n-facher Wiederholung mit g Gewinnen und $v = n - g$ Verlusten ist

$$V_N = (1 + \ell)^g \cdot (1 - L)^v V_0 \,.$$

Definieren wir die Wachstums*rate* als

$$W = \lim_{n\to\infty} \frac{1}{n} \log_2 \frac{V_n}{V_0} \,,$$

so ist

$$W = \lim_{n\to\infty} \left(\frac{g}{n} \log(1 + \ell) + \frac{v}{n} \log(1 - \ell) \right) \,.$$

Nach dem Gesetz der großen Zahlen ist im Limes mit Wahrscheinlichkeit eins $g = qn$ und $v = pn$, also

$$W = q \log(1 + \ell) + p \log(1 - \ell) \,.$$

KELLY schlägt nun vor, diese Wachstumsrate zu maximieren: Sie ist zwar auch nicht garantiert, wird aber auch und gerade für große Werte von n mit recht hoher Wahrscheinlichkeit nicht wesentlich unterschritten.

Wir wollen also W als Funktion von ℓ maximieren, wobei ℓ das abgeschlossene Einheitsintervall durchläuft. In den Intervallenden hat jeweils einer der beiden Logarithmen das Argument 0, die Wachstumsrate geht also gegen $-\infty$, wenn wir uns den Intervallenden nähern. Wegen der Stetigkeit der Funktion auf $(0,\ 1)$ muß es daher ein Maximum in diesem offenen Intervall geben; dort verschwindet

$$\frac{dW}{d\ell} = \frac{q}{1 + \ell} - \frac{p}{1 - \ell} = \frac{q(1 - \ell) - p(1 + \ell)}{1 - \ell^2} = \frac{q - p - (q + p)\ell}{1 - \ell^2} \,.$$

Da $p + q = 1$ ist, muß $\ell = q - p$ sein und damit

$$1 + \ell = 1 + q - p = 2q \quad \text{und} \quad 1 - \ell = 1 - q + p = 2p \,.$$

Im Maximum ist die Wachstumsrate daher gleich

$$\begin{aligned} W_{\max} &= q \log 2q + p \log 2p = q \log q + p \log p + (q + p) \log 2 \\ &= 1 + p \log p + q \log q \,, \end{aligned}$$

also gleich der SHANNONschen Übertragungsrate für einen Kanal mit gleicher Störungswahrscheinlichkeit p.

Für Wetten auf Ereignisse mit mehreren möglichen Ausgängen verallgemeinert KELLY seinen Ansatz wie folgt:

Wir betrachten ein Rennen, bei dem m Pferde starte; die Wahrscheinlichkeit für den Sieg des k-ten Pferds sei p_k. Vor dem Rennen können Wetten auf den Sieger abgeschlossen werden. Dazu legt der Wettanbieter für jedes der Pferde eine Quote o_k fest: Wer einen Einsatz e auf das k-te Pferd setzt, verliert diesen Einsatz, falls ein anderes Pferd gewinnt; wenn er richtig getippt hat, bekommt er das o_k-fache seines Einsatzes ausbezahlt, hat also einen Gewinn von $(o_k - 1)e$.

Um nicht gegebenenfalls alles zu verlieren, wird ein Spieler seinen Einsatz oft auf mehrere Pferde verteilen; diese Verteilung beschreiben wir durch ein *Portfolio,* d.h. einen Vektor $(b_1, \ldots, b_m)$ mit $b_k \geq 0$ für alle k und $\sum_{k=1}^{m} b_k = 1$; dabei soll b_k festlegen, daß vom Gesamteinsatz e der Teil $b_k e$ auf das k-te Pferd gesetzt wird.

Das Rennen selbst beschreiben wir durch eine Zufallsvariable X, die den Wert k annimmt, wenn das k-te Pferd siegt. Die weitere Zufallsvariable $S(X)$ gibt an, womit der Einsatz am Ende des Rennens multipliziert wird; für $X = k$ ist das $S(k) = b_k o_k$, denn nur der auf das k-te Pferd gesetzte Teil des Einsatzes wird mit o_k multipliziert; der Rest verfällt. Der Erwartungswert für $S(X)$ ist somit

$$\mathbb{E}\big(S(X)\big) = \sum_{k=1}^{m} p_k b_k o_k \,.$$

Eine möglicher Strategie zur Optimierung des Portfolios ist die Maximierung dieses Erwartungswerts; da sowohl die Zielfunktion als auch die Nebenbedingungen an die b_k linear sind, handelt es sich hier um ein Problem der linearen Optimierung, d.h. das Maximum wird in einem Eckpunkt angenommen.

Die Eckpunkte des zulässigen Bereichs sind die Punkte, in denen ein $b_k = 1$ ist, während der Rest verschwindet; für diese Portfolios ist der Erwartungswert von $S(X)$ gleich $p_k o_k$. Bei dieser Strategie setzt man also den gesamten Einsatz auf dasjenige Pferd, bei dem das Produkt $p_k o_k$ maximal wird. Für jemand, der jede Woche eine kleinere Summe setzt, deren Totalverlust er verschmerzen kann, ist das durchaus eine sinnvolle Strategie; man muß aber bedenken, daß die Wahrscheinlichkeit eines Totalverlusts mit $1 - p_k$ sehr groß sein kann: Das maximale Produkt

könnte beispielsweise einem krassen Außenseiter entsprechen, für den, da niemand mit seinem Sieg rechnet, eine große Quote o_k angeboten wird.

Wir betrachten Pferderennen hier als Einstieg in das deutlich komplexere Gebiet der Kapitalanlage; hier ist eine Strategie mit sehr wahrscheinlichem Totalverlust natürlich nicht akzeptabel, und wir müssen unser Optimierungsproblem anders stellen.

Der Ansatz, den wir hier verfolgen, geht zurück auf eine Arbeit von J.L. KELLY JR. von den (im Rahmen der Öffnung des amerikanischen Telephonmarkts inzwischen aufgeteilten) *Bell Telephone Laboratories,* die dieser im Juli 1956 in deren Forschungszeitschrift unter dem Titel *A New Interpretation of Information Rate* veröffentlichte. Er bringt Nachhaltigkeit dadurch ins Spiel, daß er von einer vielfachen Wiederholung des Rennens ausgeht. Anstelle eines einzigen Rennens, beschrieben durch eine Zufallsvariable X, betrachten wir also eine große Anzahl n von Rennen, beschrieben durch unabhängige Zufallsvariablen $X_1, \ldots, X_n$, die allesamt dieselbe Verteilungsfunktion haben wie X. Beim ersten Rennen X_1 setzen wir den Einsatz gemäß dem gewählten Portfolio, bei jedem folgenden Rennen setzen wir die Auszahlung des vorigen Rennens gemäß diesem Portfolio. Nach n Rennen ist der ursprüngliche Einsatz dann multipliziert mit

$$S_n = \prod_{i=1}^{n} S(X_i) \,.$$

Eine nachhaltige Strategie könnte darauf basieren, daß S_n als Funktion von n zumindest im Mittel möglichst schnell wachsen sollte. Strategien mit hohem Risiko eines Totalverlusts sind dadurch praktisch ausgeschlossen, denn sobald ein $S(X_i)$ verschwindet, ist $S_n = 0$ für alle $n \geq i$.

Das Wachstumsverhalten von S_n kann am besten mittels des Erwartungswerts des Logarithmus von $S(X)$ beschrieben werden:

Definition: Die Verdoppelungsrate eines Rennens mit Portfolio b und

Wahrscheinlichkeitsverteilung p ist

$$W(b,p) \underset{\text{def}}{=} \mathbb{E}\big(\log S(X)\big) = \sum_{k=1}^{m} p_k \log_2(b_k o_k) \,.$$

Lemma: S_n wächst als Funktion von n mit Wahrscheinlichkeit eins asymptotisch wie $2^{nW(b,p)}$.

Beweis: Da wir die hypothetischen Wiederholungen X_i als unabhängig annehmen, sind auch die Zufallsvariablen $\log_2 S(X_i)$ unabhängig, und sie haben allesamt dieselbe Verteilung wie $\log_2 S(X)$. Nach dem Gesetz der großen Zahl konvergiert

$$\frac{1}{n} \log_2 S(X) = \frac{1}{n} \sum_{i=1}^{n} \log_2 S(X_i)$$

daher mit Wahrscheinlichkeit eins gegen den Erwartungswert

$$\mathbb{E}\big(\log_2 S(X)\big) = W(b,p) \,,$$

der Logarithmus von S_n wächst also wie $nW(b,p)$. ∎

Wenn wir ein möglichst schnelles Wachstum von S_n wollen, müssen wir also die Verdoppelungsrate $W(b,p)$ maximieren.

Definition: Ein Portfolio b^* heißt *log-optimal,* wenn $W(b,p)$ für $b = b^*$ seinen maximalen Wert annimmt.

(Da die Bedingungen $b_k \geq 0$ und $\sum b_k = 1$ eine kompakte Teilmenge des $\mathbb{R}^m$ definieren, ist klar, daß die stetige Funktion $W(b,p)$ dort ein Maximum annehmen muß.)

Zur Berechnung eines log-optimalen Portfolios arbeiten wir wieder mit LAGRANGE-Multiplikatoren; um die Ableitungen einfach zu halten, maximieren wir anstelle von $W(b,p)$ die dazu proportionale Funktion

$$f(b) = W(b,p) \cdot \ln 2 = \sum_{k=1}^{m} p_k \ln b_k o_k \,.$$

Ihre partielle Ableitung nach b_k ist

$$\frac{\partial f(b)}{\partial b_k} = \frac{p_k}{b_k} ;$$

die Nebenbedingungsfunktion $g(b) = \sum b_k - 1$ hat partielle Ableitung eins. Im Optimum muß es daher ein $\lambda \in \mathbb{R}$ geben mit

$$\frac{p_k}{b_k} = \lambda \quad \text{oder} \quad p_k = \lambda b_k \quad \text{für alle } k .$$

Da sowohl die Summe aller p_k als auch die aller b_k eins sind, muß $\lambda = 1$ sein, d.h. $b_k = p_k$. In diesem Punkt $b^* = p$ ist

$$\begin{aligned} W(b^*, p) = W(p, p) &= \sum_{k=1}^{m} p_k \log_2 p_k o_k \\ &= \sum_{k=1}^{m} p_k \log_2 o_k + \sum_{k=1}^{m} p_k \log_2 p_k \\ &= \sum_{k=1}^{m} p_k \log_2 o_k - H(p) ; \end{aligned}$$

wir müssen zeigen, daß dies der maximal mögliche Wert für $W(b, p)$ ist. Für jedes Portfolio b gilt

$$\begin{aligned} W(b, p) &= \sum_{k=1}^{m} p_k \log_2 b_k o_k = \sum_{k=1}^{m} p_k \log_2 \left(\frac{b_k}{p_k} p_k o_k \right) \\ &= \sum_{k=1}^{m} p_k \log_2 o_k + \sum_{k=1}^{m} \log_2 p_k + \sum_{k=1}^{m} p_k \log_2 \frac{b_k}{p_k} \\ &= \sum_{k=1}^{m} p_k \log_2 o_k - H(p) - D(p\|b) = W(b^*, p) - D(p\|b) \\ &\leq W(b^*, p) , \end{aligned}$$

da die KULLBACK-LEIBLER-Distanz $D(p\|b)$ keine negativen Werte annimmt. Sie verschwindet genau dann, wenn $b = p = b^*$ ist, also liegt dort das einzige Maximum.

Das optimale Wachstum von S_n wird also erreicht, wenn wir auf jedes Pferd genau den Anteil des Einsatzes wetten, der seiner Gewinnwahrscheinlichkeit entspricht. Bei der praktischen Umsetzung dieser wie auch fast jeder anderen Strategie haben wir das Problem, daß wir die Gewinnwahrscheinlichkeiten nicht wirklich kennen; wir können sie nur schätzen anhand von früheren Rennen der beteiligten Pferde und ähnlichen Informationen. Genau das gleiche Problem hat freilich auch der Wettanbieter bei der Festlegung seiner Quoten: Auch er kennt die p_k nicht.

Nehmen wir zunächst an, er wolle seine Quoten so bestimmen, daß jeder Wetter *im Mittel* gerade seinen Einsatz zurückbekommt, daß der Wettanbieter also *im Mittel* weder einen Gewinn noch einen Verlust macht. Für einen Wetter, der einen Einsatz e auf das k-te Pferd setzt, ist die erwartete Auszahlung gleich $p_k o_k e$; bei so einer *fairen* Wette sollte also $o_k = 1/p_k$ sein. Da der Wettanbieter die p_k nicht kennt, muß er stattdessen mit einer geschätzten Wahrscheinlichkeitsverteilung $(r_1, \ldots, r_m)$ arbeiten und $o_k = 1/r_k$ setzen; entsprechend nimmt der Wetter als Portfolio *seine* Schätzung $(b_1, \ldots, b_m)$ der Wahrscheinlichkeitsverteilung. Die Verdoppelungsrate bei dieser Strategie ist

$$\begin{aligned} W(b,p) &= \sum_{k=1}^{m} p_k \log_2 b_k o_k = \sum_{k=1}^{m} p_k \log_2 \left(\frac{b_k}{p_k} \frac{p_k}{r_k} \right) \\ &= \sum_{k=1}^{m} p_k \log_2 \frac{p_k}{r_k} - \sum_{k=1}^{m} p_k \log_2 \frac{p_k}{b_k} \\ &= D(p\|r) - D(p\|b)\,, \end{aligned}$$

d.h. die Verdoppelungsrate ist positiv, wenn der Wetter die Wahrscheinlichkeitsverteilung besser geschätzt hat, und negativ, falls der Buchmacher die bessere Schätzung hatte.

(Echte Sportwetten sind natürlich nie fair; hier sind die Quoten deutlich kleiner als $1/r_k$, denn sowohl der Wettanbieter als auch das Finanzamt und gegebenenfalls der Ausrichter des Rennens wollen ein sicheren Gewinn einstreichen.)

Wer nicht professionell auf Pferde wettet, dürfte im allgemeinen keine realistische Chance haben, die Gewinnwahrscheinlichkeiten besser zu

schätzen als ein erfahrenes auf Pferdewetten spezialisiertes Wettbüro. Er könnte aber seine Chancen erhöhen, indem er den Rat von Fachleuten einholt, also zum Beispiel Spezialzeitschriften oder Rundbriefe abonniert. Da diese nicht umsonst verteilt werden, stellt sich natürlich die Frage, ob sich dieser Aufwand lohnt.

Abstrakt mathematisch betrachtet läßt sich der Wert solcher Zusatzinformation einfach berechnen: Wir kodieren die Information in einer Zufallsvariablen Y und können nun die gemeinsame Wahrscheinlichkeitsverteilung der beiden Zufallsvariablen X und Y betrachten. Ohne Zusatzinformation würden wir mit einen Portfolio $b = (b_1, \ldots, b_m)$ arbeiten, das auf unserer Schätzung des Vektors der Wahrscheinlichkeiten beruht; mit Kenntnis von Y verwenden wir stattdessen ein Portfolio $b(y) = \big(b_1(y), \ldots, b_m(y)\big)$, das auf den bedingten Wahrscheinlichkeiten $p_k(y) = p(X = k|Y = y)$ unter der Nebenbedingung $Y = y$ beruht. Die optimale Verdoppelungsrate ohne Kenntnis von Y ist, wie wir oben gesehen haben,

$$W^*(X) = \sum_{k=1}^{m} p_k \log_2 o_k - H(X)\,;$$

mit Kenntnis von Y erreichen wir

$$W^*(X|Y) = \sum_{k=1}^{m} \sum_{y} p_k(y) \log_2 p_k(y) o_k = \sum_{k=1}^{m} p_k \log_2 o_k - H(X|Y)\,.$$

Die Differenz zwischen den beiden Werten ist

$$\Delta W = W^*(X|Y) - W^*(X) = H(X) - H(X|Y) = I(X;Y)\,;$$

der Zuwachs bei der optimalen Verdoppelungsrate ist also gerade die wechselseitige Information der beiden Zufallsvariablen.

§2: Portfolio Management

Wenn man Zeitungsberichte der letzten Jahre liest, bekommt man durchaus den Eindruck, als handelten selbst große Banken am Aktienmarkt ähnlich wie Zocker, die bei Pferderennen alles auf einen Außenseiter mit minimalen Gewinnchancen setzen. Trotzdem gibt es aus Sicht eines

Mathematikers große Unterschiede zwischen einem Pferderennen und einer Börse: Während es beim Pferderennen immer nur einen Sieger gibt, kann es an der Börse durchaus vorkommen, daß an einem Tag alle gehandelten Aktien gewinnen und an einem anderen Tag alle verlieren. Dabei geht es, von eher seltenen Ausnahmen abgesehen, bei Gewinn und Verlust nicht um alles oder nichts, sondern an den meisten Börsentagen um eher moderate Kursveränderungen.

Entsprechend komplizierter muß auch unser mathematisches Modell sein: Eine einzige Zufallsvariable, die den Gewinner des Rennens beschreibt, genügt hier definitiv nicht mehr.

a) Das Modell

Wenn wir von einer Börse ausgehen, an der m Aktien gehandelt werden, brauchen wir für jede einzelne Aktie eine eigene Zufallsvariable, die deren Entwicklung beschreibt. Wir gehen der Einfachheit halber davon aus, daß wir uns für den Wert der einzelnen Aktien nur in gewissen diskreten Zeitintervallen interessieren; der Anschaulichkeit halber wird im folgenden meist von einem Börsentag die Rede sein, beim computergestützten Handel an heutigen Börsen kann es sich bei diesem Zeitintervall aber auch durchaus um eine Minute oder einen noch kleineren Zeitraum handeln.

Für jede der m Aktien betrachten wir eine Zufallsvariable X_k, die angibt, mit welchem Faktor der Kurs dieser Aktie im betrachteten Zeitraum multipliziert wird. Wir müssen realistischerweise davon ausgehen, daß wir über die Verteilungsfunktionen dieser Zufallsvariablen nur wenig wissen: Durch langfristige Beobachtung können wir zwar einige statistische Kennwerte der Verteilung schätzen; da wir aber nicht davon ausgehen können, daß die Verteilungsfunktion langfristig stabil bleibt, sind selbst diese Schätzungen von begrenztem Nutzen, und über die exakte Verteilungsfunktion werden wir nur in seltenen Fällen halbwegs zuverlässige Aussagen machen können. In diesem Paragraphen muß es daher zwangsläufig deutlich weniger konkret zugehen als im vorigen.

Trotzdem arbeiten wir wieder im wesentlichen mit denselben Begriffen:

Wir beschreiben das Geschehen an der Börse durch einen Vektor

$$X = (X_1, \ldots, X_m)$$

aus den Zufallsvariablen für die Wertentwicklung der einzelnen Aktien und wählen als Anlagestrategie wieder ein Portfolio $b = (b_1, \ldots, b_m)$, das angibt, welcher Teil des Anlagekapitals wir in welche Aktie investieren.

Vom gesamten Kapitaleinsatz K entfällt also der Teil $b_k K$ auf die k-te Aktie, und dieser wird am Ende des Börsentags mit dem Wert von X_k multipliziert. Der Wert der gesamten Anlage ist dann also

$$\sum_{k=1}^{m} b_k K X_k = K \cdot S(X) \quad \text{mit} \quad S(X) = \sum_{k=1}^{m} b_k X_k = \langle b, X \rangle \ .$$

Eine mögliche Anlagestrategie könnte darin bestehen, daß wir den Erwartungswert der Wertentwicklung $S(X)$ maximieren wollen; da

$$\mathbb{E}\big(S(X)\big) = \sum_{k=1}^{m} b_k \mathbb{E}(X_k)$$

linear in den b_k ist, würde dies bedeuten, daß wir alles Kapital in die Aktie(n) mit dem höchsten Erwartungswert investieren. Da die tatsächliche Entwicklung der Aktie nicht durch den Erwartungswert, sondern durch den Wert einer Zufallsvariable bestimmt wird, ist klar, daß so eine Strategie ihre Risiken hat.

b) log-optimale Portfolios

Eine mögliche Alternative, die von manchen Großinvestoren anscheinend auch wirklich angewendet wird, ist wieder der Ansatz von KELLY: Wir wählen die Strategie, bei der wir bei beliebig häufiger Wiederholung des Börsentags die größte Wachstumsrate erzielen. Wir betrachten also anstelle des einen Vektors X eine Folge von Vektoren $X^{(i)}$ von jeweils m Zufallsvariablen, wobei die $X^{(i)}$ voneinander unabhängig sein sollen, aber allesamt dieselbe Verteilungsfunktion F haben. Wir suchen ein Portfolio b, für das die Folge der

$$S_n = \prod_{i=1}^{n} S(X^{(i)})$$

zumindest im Mittel das größtmögliche Wachstum hat. Man beachte, daß wir bei diesem Ansatz zwar von einem festen Portfolio b ausgehen, daß dieses sich aber nicht auf die Anzahl, sondern auf den *Wert* der Aktien bezieht. Da sich dieser Wert ständig ändert, muß nach jeder „Wiederholung“ eines Börsentags der Aktienbestand neu ausbalanciert werden, damit wieder der Anteil b_k des vorhandenen Kapitals in Aktie k investiert ist.

Um in diesem Sinne optimale Portfolios zu charakterisieren, betrachten wir wieder die *Verdoppelungsrate* als Erwartungswert des Logarithmus von $S(X)$, d.h.

$$W(b,F) = \mathbb{E}\big(\log_2 S(X)\big) = \mathbb{E}\big(\log_2(\langle b, X\rangle\big) = \int \log_2 \langle b, X\rangle \; dF(x)\,.$$

Wie im Fall der Pferdewetten gilt

Satz: Mit Wahrscheinlichkeit eins wächst S_n wie $2^{nW(b,F)}$.

Auch der *Beweis* geht im wesentlichen wie dort: Nach dem Gesetz der großen Zahlen konvergiert

$$\frac{1}{n}\log_2 S_n = \frac{1}{n}\sum_{i=1}^{n} \log_2 S(X^{(i)}) = \frac{1}{n}\sum_{i=1}^{n} \log_2 \left\langle b, X^{(i)}\right\rangle$$

mit Wahrscheinlichkeit eins gegen den Erwartungswert von $\log_2 \langle b, X\rangle$, also gegen $W(b,F)$.

■

Definition: *a)* Die optimale Wachstumsrate $W^*(F)$ ist das Maximum von $W(b,F)$, wobei b alle Portfolios durchläuft.
b) Ein Portfolio b heißt *log-optimal,* wenn $W(b,F) = W^*(F)$ ist.

Da die Menge aller Portfolios kompakt ist, wissen wir, daß $W^*(F)$ existiert und daß es log-optimale Portfolios gibt; da wir F nicht kennen, können wir allerdings weder die Verdoppelungsraten $W(b,F)$ noch deren Maximalwert $W^*(F)$ wirklich ausrechnen. Trotzdem können wir einige Aussagen über diese Funktionen machen:

Lemma: *a)* $W(b, F)$ ist linear in F und konkav in b.
b) $W^*(F)$ ist konvex in F.

Beweis: a) Die Linearität in F folgt sofort aus der Linearität der Integration. Da der Logarithmus eine konkave Funktion ist, gilt für alle $\lambda \in [0,\ 1]$

$$\log_2 \langle (1-\lambda)b_1 + \lambda b_2, X\rangle \geq (1-\lambda)\log_2 \langle b_1, X\rangle + \lambda \log_2 \langle b_2, X\rangle \ ;$$

aus der Monotonie der Integration folgt damit auch die Konkavität von $W(b, F)$ in b.

b) F_1 und F_2 seien zwei Verteilungsfunktionen, und $b^*(F_1), b^*(F_2)$ seien log-optimale Portfolios dazu. Dann ist für jedes $\lambda \in [1,\ 0]$ auch $(1-\lambda)F_1 + \lambda F_2$ eine Verteilungsfunktion; das log-optimale Portfolio dazu sei $b^*\big((1-\lambda)F_1 + \lambda F_2\big)$. Dann ist

$$W^*\big((1-\lambda)F_1 + \lambda F_2\big) = W\Big(b^*\big((1-\lambda)F_1 + \lambda F_2\big), (1-\lambda)F_1 + \lambda F_2\Big)$$

$$= (1-\lambda)W\Big(b^*\big((1-\lambda)F_1+\lambda F_2\big), F_1\Big) + \lambda W\Big(b^*\big((1-\lambda)F_1+\lambda F_2\big), F_2\Big)$$

$$\leq (1-\lambda)W\big(b^*(F_1), F_1\big)+\lambda W(\big(b^*(F_2), F_2) = (1-\lambda)W^*(F_1)+\lambda W^*(F_2)\,,$$

da $b^*(F_1)$ für F_1 und $b^*(F_2)$ für F_2 optimal ist. ∎

Im Gegensatz zur Situation bei den Pferdewetten gibt es hier natürlich keinen Grund für die Annahme, es gäbe nur ein log-optimales Portfolio; es kann eine ganze Reihe davon geben. Wir können allerdings zeigen

Lemma: Die Menge aller log-optimaler Portfolios zu einer festen Verteilungsfunktion F ist konvex.

Beweis: b und b' seien zwei log-optimale Portfolios. Wegen der Konkavität von $W(b, F)$ in b ist dann für jedes $\lambda \in [0,\ 1]$

$$\begin{aligned} W\big((1-\lambda)b_1 + \lambda b_2\big) &\geq (1-\lambda)W(b_1, F) + \lambda W(b_2, F) \\ &= (1-\lambda)W^*(F) + \lambda W^*(F) = W^*(F)\,. \end{aligned}$$

Da W^* nach Definition die maximal erreichbare Wachstumsrate ist, muß also $W\big((1-\lambda)b_1 + \lambda b_2\big) = W^*(F)$ sein, d.h. auch $(1-\lambda)b_1 + \lambda b_2$ ist ein log-optimales Portfolio. ∎

c) Eine erste Charakterisierung log-optimaler Portfolios

Da wir nur wenig über die Verteilungsfunktion F wissen, kann uns auch ein analytischer Ansatz kein konkretes Gleichungssystem liefern, anhand dessen wir ein log-optimales Portfolio berechnen könnten. Zusammen mit der Konkavität von $W(b, F)$ in b liefert er aber ein nützliches Kriterium zur Charakterisierung log-optimaler Portfolios:

Satz: *a)* Ein Portfolio $b^* = (b_1^*, \ldots, b_m^*)$ ist genau dann log-optimal bezüglich der Verteilungsfunktion F, wenn für jedes andere Portfolio b gilt:

$$\mathbb{E}\left(\frac{\langle b, X\rangle}{\langle b^*, X\rangle}\right) \leq 1 \,.$$

b) Ein Portfolio $b^* = (b_1^*, \ldots, b_m^*)$ ist genau dann log-optimal bezüglich der Verteilungsfunktion F, wenn

$$\mathbb{E}\left(\frac{X_k}{\langle b^*, X\rangle}\right) \leq 1 \qquad \text{für alle } k \,.$$

c) Ist für ein log-optimales Portfolio b^* die k-te Komponente b_k^* von null verschieden, so ist sogar

$$\mathbb{E}\left(\frac{X_k}{\langle b^*, X\rangle}\right) = 1 \,.$$

Beweis: a) Wie immer, wenn wir differenzieren müssen, betrachten wir die Wachstumsrate bezüglich des natürlichen Logarithmus, also die Funktion

$$V(b) = W(b, F) \cdot \ln 2 = \mathbb{E}\big(\ln \langle b, X\rangle\big) \,.$$

Da auch diese Funktion konkav ist, hat sie genau dann ein Maximum in einem Punkt b^* des zulässigen Bereichs, wenn ihre Richtungsableitung entlang der Strecke von b^* zu irgendeinem anderen Portfolio b stets kleiner oder gleich Null ist. (Wegen der Konvexität der Menge aller Portfolios liegt diese Strecke im zulässigen Bereich.)

Die Verbindungsstrecke besteht aus den Punkten $b_\lambda = (1-\lambda)b^* + \lambda b$ mit $\lambda \in [0,\, 1]$; da die Bildung des Erwartungswerts eine lineare Operation

ist und mit Grenzwertbildung kommutiert, ist

$$\left.\frac{dV(b_\lambda)}{d\lambda}\right|_{\lambda=0^+} = \left.\frac{d}{d\lambda}\mathbb{E}\big(\ln\langle b_\lambda, X\rangle\big)\right|_{\lambda=0^+}$$
$$= \lim_{\lambda\searrow 0}\mathbb{E}\left(\frac{\ln\langle b_\lambda, X\rangle - \ln\langle b^*, X\rangle}{\lambda}\right) = \lim_{\lambda\searrow 0}\mathbb{E}\left(\frac{1}{\lambda}\ln\frac{\langle b_\lambda, X\rangle}{\langle b^*, X\rangle}\right)$$
$$= \lim_{\lambda\searrow 0}\frac{1}{\lambda}\mathbb{E}\left(\ln\frac{(1-\lambda)\langle b^*, X\rangle + \lambda\langle b, X\rangle}{\langle b^*, X\rangle}\right)$$
$$= \mathbb{E}\left(\lim_{\lambda\searrow 0}\frac{1}{\lambda}\ln\left(1+\lambda\left(\frac{\langle b, X\rangle}{\langle b^*, X\rangle}-1\right)\right)\right)$$

Aus der TAYLOR-Entwicklung von

$$\ln(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \cdots$$

oder auch aus dem Mittelwertsatz der Differentialrechnung folgt, daß

$$\lim_{\lambda\to 0}\frac{\ln(1+\lambda x)}{\lambda} = x$$

ist, also können wir die Ableitung weiter ausrechnen als

$$\left.\frac{dV(b_\lambda)}{d\lambda}\right|_{\lambda=0^+} = \mathbb{E}\left(\frac{\langle b, X\rangle}{\langle b^*, X\rangle}\right) - 1\,.$$

b^* ist somit genau dann ein log-optimales Portfolio, wenn dies für alle Portfolios b kleiner oder gleich Null ist, und das wiederum ist äquivalent zur Ungleichung $\mathbb{E}\left(\frac{\langle b,X\rangle}{\langle b^*,X\rangle}\right) \le 1$.

b) Da alle $b_k \ge 0$ sind und

$$\mathbb{E}\left(\frac{\langle b, X\rangle}{\langle b^*, X\rangle}\right) - 1 = \sum_{k=1}^{m} b_k\mathbb{E}\left(\frac{X_k}{\langle b^*, X\rangle}\right) - 1$$
$$= \sum_{k=1}^{m} b_k\left(E\left(\frac{X_k}{\langle b^*, X\rangle}\right) - 1\right)$$

ist, gilt die Bedingung aus *a)* genau dann, wenn alle $E\big(\frac{X_k}{\langle b^*,X\rangle}\big) \le 1$ sind.

c) Falls b_k^* nicht verschwindet, betrachten wir jenes Portfolio b, das das gesamte Kapital in die k-te Aktie investiert. b^* ist dann ein innerer Punkt auf dem Durchschnitt der Geraden durch b^* und b mit der Menge aller Portfolios; daher hat $V(b)$ auf diesem Durchschnitt genau dann ein Maximum in b^*, wenn dort die Richtungsableitung sogar verschwindet, d.h. $\mathbb{E}\left(\frac{X_k}{\langle b^*, X\rangle}\right)$ muß gleich eins sein.

■

Die Bedingung aus *a)* sagt insbesondere, daß log-optimale Portfolios auch bezüglich des Erwartungswerts des Quotienten der Gewinnentwicklung optimal sind; sie erfüllen also auch ein kurzfristiges Optimalitätskriterium.

Nach der hier betrachteten Strategie wird nach jedem Börsentag das vorhandene Kapital so reinvestiert, daß der Anteil b_k auf die k-te Aktie entfällt. Welcher Teil des Kapitals am Ende des Börsentags in welcher Aktie steckt, hängt natürlich von der konkreten Entwicklung am jeweiligen Tag ab, aber für ein log-optimales Portfolio b^* können wir zumindest den Erwartungswert berechnen: Das gesamte Kapital nach Handelsschluß ist das $\langle b^*, X\rangle$-fache des Ausgangskapitals K, und aus dem Investment $b_k^* K$ in die k-te Aktie wurde ein Betrag von $b_k^* K X_k$. Der Anteil der k-ten Aktie ist daher $b_k^* X_k / \langle b^*, X\rangle$ mit Erwartungswert

$$\mathbb{E}\left(\frac{b_k^* X_k}{\langle b^*, X\rangle}\right) = b_k^* \mathbb{E}\left(\frac{X_k}{\langle b^*, X\rangle}\right) = b_k^*$$

unabhängig davon, ob b_k^* verschwindet oder nicht. Zumindest was die Erwartungswerte betrifft, bleiben die Anteile also stabil.

d) Asymptotische Optimalität

Wenn wir von einer Börse ausgehen, bei der die Verteilungsfunktion F über einen längeren Zeitraum fest bleibt, können wir die KELLY-Strategie statt auf hypothetische Wiederholungen eines Börsentags auch auf die Folge der Börsentage anwenden.

Da die Verteilungsfunktion konstant bleibt, ist auch das log-optimale Portfolio jeden Tag das gleiche; wir investieren also nach einem festen log-optimalen Portfolio b^*, wobei wir zu Beginn eines jedes Börsentags

dafür sorgen, daß unabhängig von der Kursentwicklung des Vortags wieder der Anteil b_k^* des Anlagekapitals in Aktie k investiert wird.

Diese Strategie erscheint recht unflexibel; wenn wir stattdessen das Portfolio jeden Tag neu festlegen in Abhängigkeit von der bisherigen Kursentwicklung der Aktien, können wir möglicherweise ein besseres Ergebnis erzielen.

Definition: Eine *kausale Anlagestrategie* ist eine Folge von Abbildungen

$$b^{(i)}\colon \begin{cases} \mathbb{R}^{m(i-1)} \to \mathbb{R}^m \\ (x^{(1)}, \ldots, x^{(i-1)}) \mapsto b^{(i)}(x^{(1)}, \ldots, x^{(i-1)}) \end{cases},$$

die in Abhängigkeit von den Kurs Entwicklungen $X^{(j)} = x^{(j)}$ für $j < i$ das Portfolio für den i-ten Börsentag festlegt.

Eine einfache Abschätzung zeigt aber, daß man trotz der größeren Flexibilität mit einer solchen Strategie zumindest im Mittel keine besseren Ergebnisse erzielt als mit der log-optimalen Strategie:

Lemma: Bezeichnet $S_n = \prod_{i=1}^n S(X^{(i)})$ die Wertentwicklung nach n Tagen für eine kausale Anlagestrategie, so ist $\mathbb{E}(\log_2 S_n) \leq nW^*$, wobei W^* die Verdoppelungsrate der log-optimalen Strategie ist.

Beweis: Der Erwartungswert von $\log_2 S_n$ ist höchstens gleich

$$\max_{b^{(1)},\ldots,b^{(n)}} \mathbb{E}(\log S_n) = \max_{b^{(1)},\ldots,b^{(n)}} \mathbb{E}\left(\sum_{i=1}^n \log_2 \left\langle b^{(i)}, X^{(i)} \right\rangle\right)$$

$$= \sum_{i=1}^n \max_{b^{(i)}} \mathbb{E}\left(\log_2 \left\langle b^{(i)}, X^{(i)} \right\rangle\right) = nW^* .$$

∎

Nun wissen wir freilich, daß der Erwartungswert allein noch nicht viel aussagt; das gerade bewiesene Lemma schließt nicht aus, daß wir mit einer geeigneten kausalen Strategie vielleicht doch mit hoher Wahrscheinlichkeit besser fahren als mit der log-optimalen. Um zu sehen, daß auch das nicht der Fall ist, benötigen wir zunächst zwei Aussagen aus der Wahrscheinlichkeitstheorie, als erstes einen einfachen Spezialfall der Ungleichung von MARKOV:

Lemma: Ist X eine Zufallsvariable, die nichtnegative reelle Zahlen als Werte annimmt, so ist für jedes $a > 0$

$$p(X \geq a) \leq \frac{\mathbb{E}(X)}{a} .$$

Beweis: $\chi_a(x)$ sei die charakteristische Funktion der Menge aller reeller Zahlen größer oder gleich a. Dann ist, wenn P das Wahrscheinlichkeitsmaß bezeichnet,

$$p(X \geq a) = \int \chi_a(x)\, dP \leq \int \chi_a(x) \frac{x}{a}\, dP \leq \int \frac{x}{a}\, dP = \frac{\mathbb{E}(X)}{a} .$$

■

Lemma von Borel und Cantelli: $E_1, E_2, \ldots$ sei eine Folge von Ereignissen derart, daß $\sum_{i=1}^{\infty} p(E_i)$ konvergiert. Dann ist die Wahrscheinlichkeit für das Eintreten von unendlich vielen dieser Ereignisse gleich Null.

Beweis: Die Wahrscheinlichkeit dafür, daß irgendein Ereignis E_k mit $k \geq n$ eintritt, ist höchstens $p_n = \sum_{i=n}^{\infty} p(E_i)$; wegen der vorausgesetzten Konvergenz der Summe bilden die p_n eine Nullfolge. Falls unendlich viele der Ereignisse E_i auftreten, tritt insbesondere für jedes n mindestens eines mit $i \geq n$ ein; die Wahrscheinlichkeit für das Eintreten unendlich vieler der Ereignisse ist somit für jedes n kleiner oder gleich p_n. Damit muß diese Wahrscheinlichkeit gleich Null sein. ■

EMILE BOREL (1871–1956) wurde in Saint-Affrique in den Pyrenäen als Sohn eines protestantischen Geistlichen geboren. Ab 1889 studierte er Mathematik an der Ecole Normale in Paris, wo er 1893 promovierte. Danach bekam er zunächst eine Stelle als *maître de conférence* an der Universität von Lille, drei Jahre später an der Ecole Normale Supérieure; von 1899 bis 1902 lehrte er am Collège de France. 1909 richtete die Sorbonne speziell für ihn einen Lehrstuhl ein, den er bis 1941 innehatte. Ab 1924 war er auch politisch aktiv als Abgeordneter der Nationalversammlung (1924–1936) und als Marineminister (1925–1940). Während des zweiten Weltkriegs kämpfte er für die Résistance. Seine mathematischen Arbeiten kommen aus fast allen Teilgebieten der Mathematik; besonders bekannt sind seine Beiträge

zur Maß- und Wahrscheinlichkeitstheorie sowie zur Theorie reeller und komplexer Funktionen.

FRANCESCO PAOLO CANTELLI (1875–1966) wurde in Palermo geboren und studierte an der dortigen Universität Mathematik. Neben seinem Studium arbeitete er auch am dortigen Observatorium und veröffentlichte verschiedene Arbeiten über Astronomie. Von 1903 bis 1923 arbeitete er als Aktuar für eine Versicherung und beschäftigte sich mit Anwendungen der Wahrscheinlichkeitstheorie; danach lehrte er Finanz- und Versicherungsmathematik zunächst an der Universität von Catania, ab 1925 in Neapel und ab 1931 bis zu seiner Emeritierung 1951 in Rom. Neben vielen anderen Arbeiten zur Wahrscheinlichkeitstheorie bewies er unter anderem auch das starke Gesetz der großen Zahlen.

Außerdem sei noch an zwei Begriffe aus der Analysis erinnert:

Definition: *a)* Eine reelle Zahl x heißt *Häufungspunkt* der reellen Zahlenfolge $(x_n)_{n\in\mathbb{N}}$, wenn es zu jedem $\varepsilon > 0$ unendlich viele Folgenglieder x_n gibt mit $|x - x_n| < \varepsilon$.
b) Der kleinste Häufungspunkt einer Folge wird als *Limes inferior* $\underline{\lim}_{n\to\infty} x_n$ bezeichnet, der größte als *Limes superior* $\overline{\lim}_{n\to\infty} x_n$.

Damit können wir nun beweisen, daß keine Strategie langfristig wesentlich besser sein kann als die log-optimale:

Satz: $X^{(1)}, X^{(2)}, \ldots$ sei eine Folge unabhängiger Vektoren von Zufallsvariablen, die allesamt Verteilungsfunktion F haben. Das log-optimale Portfolio dazu sei b^*, und $b^{(1)}, b^{(2)}, \ldots$ sei irgendeine kausale Anlagestrategie. Für die beiden Wertentwicklungen $S_n^* = \prod_{i=1}^n \left\langle b^*, X^{(i)} \right\rangle$ und $S_n = \prod_{i=1}^n \left\langle b^{(i)}, X^{(i)} \right\rangle$ ist dann mit Wahrscheinlichkeit eins

$$\overline{\lim_{n\to\infty}} \frac{1}{n} \log_2 \frac{S_n}{S_n^*} \leq 0\,.$$

Beweis: Wie wir aus Abschnitt *c)* wissen, ist der Erwartungswert von S_n/S_n^* kleiner oder gleich eins; nach der Ungleichung von MARKOV ist daher

$$p(S_n > n^2 S_n^*) = p\left(\frac{S_n}{S_n^*} > n^2\right) < \frac{1}{n^2}\,.$$

Da $\sum_{n=1}^{\infty} \frac{1}{n^2}$ konvergiert, sagt uns das Lemma von BOREL und CANTELLI, daß mit Wahrscheinlichkeit null unendlich viele der Ungleichungen

$$\frac{S_n}{S_n^*} > n^2 \quad \text{oder} \quad \frac{1}{n} \log_2 \frac{S_n}{S_n^*} > \frac{2 \log_2 n}{n}$$

erfüllt sind. Es gibt daher mit Wahrscheinlichkeit eins ein $N \in \mathbb{N}$, so daß für alle $n \geq N$ gilt

$$\frac{1}{n} \log \frac{S_n}{S_n^*} < \frac{2 \log_2 n}{n} .$$

Da rechts eine Nullfolge steht, kann der Limes superior nicht positiv sein.

■

e) Der Einfluß zusätzlicher Information

Nicht nur bei Pferderennen, sondern auch an der Börse wird viel mehr oder weniger nützliche Zusatzinformation angeboten. Um deren Wert abzuschätzen, betrachten wir zunächst, wie sich die Wachstumsrate eines log-optimalen Portfolios ändert, wenn wir von einer falschen Verteilungsfunktion ausgehen. Bei Pferderennen hatten wir gesehen, daß dies mit der KULLBACK-LEIBLER-Distanz der Wahrscheinlichkeitsverteilungen zusammenhängt; da wir hier kontinuierliche Zufallsvariablen haben, müssen wir diese erst definieren:

Definition: *a)* Die KULLBACK-LEIBLER-Distanz zwischen zwei Wahrscheinlichkeitsdichten f und g ist

$$D(f \| g) = \int f(x) \log_2 \frac{f(x)}{g(x)} \, dx .$$

b) Die wechselseitige Information zweier Zufallsvariablen X und Y mit den Wahrscheinlichkeitsdichten f_X und f_Y sowie der gemeinsamen Wahrscheinlichkeitsdichte f ist

$$I(X;Y) = \int f(x,y) \log_2 \frac{f(x,y)}{f_X(x) f_Y(y)} \, dx \, dy .$$

Satz: $f(x_1, \ldots, x_m)$ sei die Wahrscheinlichkeitsdichte zum Vektor X der Zufallsvariablen $X_1, \ldots, X_m$, und b_f sei das log-optimale Portfolio dazu. $g(x_1, \ldots .x_m)$ sei eine weitere Wahrscheinlichkeitsdichte, und b_g sei das log-optimale Portfolio zu g. Dann ist

$$\Delta W = W(b_f, F) - W(b_g, F) \leq D(f\|g)\,.$$

Beweis: Nach Definition ist

$$\begin{aligned}
\Delta W &= \int f(x) \log_2 \left\langle b_f, x \right\rangle \, dx - \int f(x) \log_2 \left\langle b_g, x \right\rangle \, dx \\
&= \int f(x) \log_2 \frac{\left\langle b_f, x \right\rangle}{\left\langle b_g, x \right\rangle} \, dx \\
&= \int f(x) \log_2 \left(\frac{\left\langle b_f, x \right\rangle}{\left\langle b_g, x \right\rangle} \frac{g(x)}{f(x)} \frac{f(x)}{g(x)} \right) dx \\
&= \int f(x) \log_2 \left(\frac{\left\langle b_f, x \right\rangle}{\left\langle b_g, x \right\rangle} \frac{g(x)}{f(x)} \right) dx + D(f\|g)
\end{aligned}$$

Da f eine Wahrscheinlichkeitsdichte ist, können wir die Ungleichung von JENSEN auf das letzte Integral anwenden und den Logarithmus nach vorne ziehen; wir erhalten

$$\begin{aligned}
\int f(x) \log_2 \left(\frac{\left\langle b_f, x \right\rangle}{\left\langle b_g, x \right\rangle} \frac{g(x)}{f(x)} \right) dx &\leq \log_2 \int f(x) \frac{\left\langle b_f, x \right\rangle}{\left\langle b_g, x \right\rangle} \frac{g(x)}{f(x)} \, dx \\
&= \log_2 \int g(x) \frac{\left\langle b_f, x \right\rangle}{\left\langle b_g, x \right\rangle} \, dx \leq \log 1 = 0\,,
\end{aligned}$$

da b_g das log-optimale Portfolio zur Wahrscheinlichkeitsdichte g ist. Somit ist $\Delta W \leq D(f\|g)$, wie behauptet. ∎

Diesen Satz wollen wir nun anwenden auf den Fall, daß wir zusätzliche Informationen über die Entwicklung der Börse haben. Diese Information beschreiben wir wieder durch eine Zufallsvariable Y. Die gemeinsame Wahrscheinlichkeitsverteilung von X und Y sei durch die Wahrscheinlichkeitsdichte $f(x, y)$ gegeben, f_X und f_Y seien die Wahrscheinlichkeitsdichten zu X und Y allein. Ohne Zusatzinformation definieren wir

das log-optimale Portfolio anhand von f_X, mit der Zusatzinformation $Y = y$ verwenden wir stattdessen f mit zweitem Argument y.

Satz: Der Anstieg ΔW der Verdoppelungsrate durch die in Y kodierte Zusatzinformation ist höchstens gleich $I(X;Y)$.

Beweis: Wenn wir das log-optimale Portfolio auf einen konkreten Wert $Y = y$ abstimmen, steigt die Verdoppelungsrate nach dem vorigen Satz höchstens um

$$D\big(f(x|X=y) \,\|\, f(x)\big) = \int_x f(x|Y=y) \log_2 \frac{f(x|Y=y)}{f_X(x)}\, dx\,.$$

ΔW ist das mit f_Y gewichtete Mittel über diese Anstiege, also kleiner oder gleich

$$\begin{aligned}
&\int_y f_Y(y) \int_x f(x|Y=y) \log_2 \frac{f(x|Y=y)}{f_X(x)}\, dx\, dy \\
&= \int_y \int_x f_Y(y) f(x|Y=y) \log_2 \left(\frac{f(x|Y=y)}{f_X(x)} \frac{f_Y(y)}{f_Y(y)} \right) dx\, dy \\
&= \int_y \int_x f(x,y) \log_2 \frac{f(x,y)}{f_X(x) f_Y(y)}\, dx\, dy = I(X;Y)\,.
\end{aligned}$$

∎

Man beachte, daß der Zuwachs hier im Gegensatz zum Fall der Pferdewetten nicht gleich der wechselseitigen Information sein muß, sondern auch kleiner ausfallen kann.

f) Verallgemeinerung auf stationäre Märkte

Bislang sind wir davon ausgegangen, daß die verschiedenen Börsentage (oder auch die hypothetischen Wiederholungen eines Börsentags) durch voneinander unabhängige Zufallsvariablen beschrieben werden. In letzter Konsequenz bedeutet diese Annahme, daß sich die Börse am Tag nach einem großen Crash verhält, als sei nichts geschehen, daß es keine Gewinnmitnahmen nach Höhenflügen einer Aktie gibt, und so weiter.

Für den Rest dieses Paragraphen wollen wir daher unser Modell erweitern und die Kursentwicklung beschreiben durch einen stochastischen Prozess $\mathcal{X} = X^{(1)}, X^{(2)}, \ldots$ aus Zufallsvariablen $X^{(i)}$ mit Werten in $\mathbb{R}^m$,

wobei die k-te Komponente von $X^{(i)}$ angibt, wie sich die Aktie k am i-ten Börsentag entwickelt, d.h. mit welchem Faktor ihr Wert im Laufe des Tages multipliziert wird. Dieser stochastische Prozess soll stationär sein, insbesondere haben also weiterhin alle $X^{(i)}$ dieselbe Wahrscheinlichkeitsverteilung F; wir gehen aber nicht mehr davon aus, daß die Wertentwicklungen der einzelnen Tage unabhängig voneinander sind.

Für die Anlage betrachten wir weiterhin kausale Strategien, d.h. das Portfolio $b^{(i)}$ am i-ten Tag hängt ab von den Werten der $X^{(j)}$ mit $j < i$. Um herauszustellen, daß $b^{(i)}$ eine Funktion dieser Zufallsvariablen ist, schreiben wir gelegentlich auch $b^{(i)}(X^{(1)}, \ldots, X^{(i-1)})$; den Funktionswert für konkrete Werte $X^{(j)} = x^{(j)}$ bezeichnen wir mit $b(x^{(1)}, \ldots, x^{(i-1)})$.

Auch in der neuen Situation wollen wir den Erwartungswert für die langfristige Wertentwicklung der Anlage maximieren, also den Erwartungswert des Logarithmus von

$$S_n = \prod_{i=1}^{n} \left\langle b^{(i)}(X^{(1)}, \ldots, X^{(i-1)}), X^{(i)} \right\rangle \ .$$

Das Maximum dieses Erwartungswert über alle kausalen Anlagestrategien ist die Summe über die maximalen Erwartungswerte für die einzelnen Tage; wir wählen also für jeden Tag dasjenige Portfolio, von dem wir das maximale logarithmische Wachstum erwarten, d.h. das log-optimale Portfolio. Da die Zufallsvariablen $X^{(i)}$ nicht mehr als unabhängig vorausgesetzt sind, dürfen wir allerdings nicht mehr einfach jeden Tag das log-optimale Portfolio zur Wahrscheinlichkeitsverteilung F nehmen, denn wie sich $X^{(i)}$ entwickelt hängt ja ab von den Entwicklungen der Vortage. Das log-optimale Portfolio $b^{*(i)}(x^{(1)}, \ldots, x^{(i-1)})$ muß daher bestimmt werden über die bedingte Wahrscheinlichkeitsverteilung unter der Voraussetzung, daß $X^{(1)} = x^{(1)}, \ldots, X^{(i-1)} = x^{(i-1)}$ ist. Den entsprechenden Erwartungswert für den i-ten Börsentag bezeichnen wir als die bedingte Verdoppelungsrate dieses Tages und schreiben sie als

$$W^*(X^{(i)}|x^{(1)}, \ldots, x^{(i-1)}) =$$
$$\mathbb{E}\Big(\log_2 \left\langle b^{*(i)}(x^{(1)}, \ldots, x^{(i-1)}), X^{(i)} \right\rangle \ \Big|\ X^{(1)} = x^{(1)}, \ldots, X^{(i-1)} = x^{(i-1)}\Big) \ .$$

Der Erwartungswert des Logarithmus von $S(X^{(i)})$ ist der gewichtete Mittelwert $W^*(X^{(i)}|X^{(1)}, \ldots, X^{(i-1)})$ dieser bedingten Verdoppelungs-

raten über alle möglichen Entwicklungen der ersten $i-1$ Tage, und für den Erwartungswert des Logarithmus von S_n erhalten wir die Kettenregel

$$W^*(X^{(1)},\ldots,X^{(n)}) = \mathbb{E}(\log_2 S_n) = \sum_{i=1}^{n} W^*(X^{(i)}|X^{(1)},\ldots,X^{(i-1)})\,.$$

In Analogie zur Entropierate eines stochastischen Prozesses sagen wir

Definition: Die optimale Wachstumsrate ist

$$W^*_\infty \underset{\text{def}}{=} \lim_{n\to\infty} \frac{W^*(X^{(1)},\ldots,X^{(n)})}{n}\,,$$

sofern dieser Grenzwert existiert.

Genau wie im Falle der Entropierate können wir auch hier zeigen

Satz: In einen stationären Markt existiert die Wachstumsrate und ist gleich $\lim_{n\to\infty} W^*(X^{(n)}|X^{(1)},\ldots,X^{(n-1)})$.

Beweis: Da die Wachstumsrate bei mehr Information nicht kleiner werden kann und wir einen stationären Prozess vorausgesetzt haben, ist

$$\begin{aligned} W^*(X^{(n+1)}|X^{(1)},\ldots,X^{(n)}) &\geq W^*(X^{(n+1)}|X^{(2)},\ldots,X^{(n)}) \\ &= W^*(X^{(n)}|X^{(1)},\ldots,X^{(n-1)})\,, \end{aligned}$$

die Folge der $W^*(X^{(n)}|X^{(1)},\ldots,X^{(n-1)})$ ist also monoton wachsend. Somit ist sie entweder konvergent oder divergiert bestimmt gegen $+\infty$. Da nach der Kettenregel

$$\frac{W^*(X^{(1)},\ldots,X^{(n)})}{n} = \frac{1}{n}\sum_{i=1}^{n} W^*(X^{(i)}|X^{(1)},\ldots,X^{(i-1)})$$

ist, hat die linke Seite nach der Mittelwertsatz von CESÀRO *(Kap. 1, §4c), Schritt 2)* denselben Grenzwert. ∎

Um auch bei stationären Märkten die log-optimale Strategie mit anderen Anlagestrategien vergleichen zu können, müssen wir uns an einen Begriff aus der Stochastik erinnern:

Definition: $\mathcal{Y} = Y_1, Y_2, \ldots$ und $\mathcal{Z} = Z_1, Z_2, \ldots$ seien zwei stochastische Prozesse. $\mathcal{Z}$ heißt *Martingal* bezüglich $\mathcal{Y}$, wenn für alle $k < \ell$ gilt $\mathbb{E}(Z_\ell | Y_1, \ldots, Y_k) = Z_k$. Er heißt *Supermartingal,* wenn $\mathbb{E}(Z_\ell | Y_1, \ldots, Y_k) \leq Z_k$ ist; entsprechend heißt er *Submartingal,* falls $\mathbb{E}(Z_\ell | Y_1, \ldots, Y_k) \geq Z_k$.

(Tatsächlich betrachtet man in der Stochastik Martingale meist allgemeiner bezüglich einer beliebigen Filtration auf dem Wahrscheinlichkeitsraum; für uns genügt aber dieser Spezialfall.)

Satz: $\mathcal{X}$ sei ein stationärer stochastischer Prozess und S_n^* beschreibe die Wertentwicklung der ersten n Tage einer Anlage bezüglich der bedingt log-optimalen Strategie, S_n die bezüglich irgendeiner beliebigen kausalen Strategie. Dann ist die Folge $\mathcal{Q}$ der Quotienten S_n / S_n^* ein positives Supermartingal bezüglich $\mathcal{X}$.

Zum *Beweis* müssen wir den Erwartungswert

$$\mathbb{E}\left(\frac{S_{n+1}}{S_{n+1}^*} \middle| X^{(1)}, \ldots, X^{(n)} \right)$$

abschätzen. Da wir bei Kenntnis von $X^{(1)}, \ldots, X^{(n)}$ die Wertentwicklungen S_n und S_n^* kennen, können wir diese vor den Erwartungswert ziehen und erhalten

$$\mathbb{E}\left(\frac{S_{n+1}}{S_{n+1}^*} \middle| X^{(1)}, \ldots, X^{(n)} \right) = \mathbb{E}\left(\frac{\left\langle b^{(n+1)}, X^{(n+1)} \right\rangle S_n}{\left\langle b^{*(n+1)}, X^{(n+1)} \right\rangle S_n^*} \middle| X^{(1)}, \ldots, X^{(n)} \right)$$

$$= \frac{S_n}{S_n^*} \mathbb{E}\left(\frac{\left\langle b^{(n+1)}, X^{(n+1)} \right\rangle}{\left\langle b^{*(n+1)}, X^{(n+1)} \right\rangle} \middle| X^{(1)}, \ldots, X^{(n)} \right) \leq \frac{S_n}{S_n^*},$$

denn wie wir in Abschnitt *c)* gesehen haben, ist der letzte Erwartungswert für das log-optimale Portfolio kleiner oder gleich eins. Induktiv folgt die entsprechende Aussage für Index $n + k$ statt $n + 1$ und damit die Supermartingaleigenschaft.

■

Nach dem Martingalkonvergenzsatz von JOSEPH DOOB (1910–2004) folgt aus der Tatsache, daß die Folge der S_n / S_n^* ein Supermartingal ist, daß diese Folge gegen eine Zufallsvariable konvergiert deren Erwartungswert höchstens gleich dem von S_1 / S_1^* ist, also kleiner

oder gleich eins. Aus KOLMOGOROVs Verallgemeinerung der MARKOV-Ungleichung folgt daraus wiederum, daß für alle $t > 1$ gilt

$$p\left(\sup_n \frac{S_n}{S_n^*} \geq t\right) \leq \frac{1}{t}\,.$$

Die Wahrscheinlichkeit, daß irgendeine andere Strategie langfristig dramatisch bessere Ergebnisse liefert als die bedingte log-optimale ist somit relativ gering.

Kurzfristig freilich gibt es Fälle, in denen andere Strategien mit hoher Wahrscheinlichkeit zumindest etwas bessere Ergebnisse liefern als die log-optimale: An einer „Börse“ mit nur zwei Aktien und

$$X = (X_1, X_2) = \begin{cases} \left(1, \frac{1}{1-\varepsilon}\right) & \text{mit Wahrscheinlichkeit } 1-\varepsilon \\ (1,0) & \text{mit Wahrscheinlichkeit } \varepsilon \end{cases}$$

ist für das Portfolio $b = (1,0)$

$$\mathbb{E}\left(\frac{X_1}{\langle b, X\rangle}\right) = \mathbb{E}\left(\frac{X_1}{X_1}\right) = 1 \quad \text{und}$$
$$\mathbb{E}\left(\frac{X_2}{\langle b, X\rangle}\right) = \mathbb{E}\left(\frac{X_2}{X_1}\right) = \frac{1-\varepsilon}{1-\varepsilon} = 1\,,$$

nach dem Kriterium aus Abschnitt *c)* ist b also log-optimal. Trotzdem erzielt man mit dem Portfolio $(0,1)$ mit Wahrscheinlichkeit $1-\varepsilon$ ein besseres Ergebnis. Nach n Börsentagen ist allerdings die Wahrscheinlichkeit dafür, daß man mit dieser Strategie noch nicht pleite gegangen ist, gleich $(1-\varepsilon)^n$, was für hinreichend große n eine ziemlich kleine Zahl ist.

Zusammenfassend läßt sich sagen, daß log-optimale Portfolios relativ sicher sind und auf lange Sicht die besten Wachstumsraten liefern; mit relativ hoher Wahrscheinlichkeit – aber keinesfalls sicher! – liefern sie zumindest langfristig auch ein gutes Ergebnis. Wie allerdings unter anderem der Wirtschaftsnobelpreisträger von 1979 PAUL A. SAMUELSON (1915–2009) mehrfach gegen sie argumentierte, hat jeder Anleger seine eigenen Vorstellungen vom Umgang mit Risiken, so daß sie nicht unbedingt *die* Anlagestrategie für jedermann sind. *Risikofrei* sind sie definitiv nicht; wie SAMUELSON in seiner Arbeit *Why we should not make mean*

log of wealth big though years to act are long (Journal of Banking and Finance **3** (1979), 305–307) sagt (Sein N ist unser n):

> When you lose – and you *sure can* lose – with N large, you can lose real big. Q.E.D.

§3: Universelle Portfolios

Die log-optimalen Portfolios des letzten Paragraphen waren definiert in Bezug auf die Verteilungsfunktion für die Wertentwicklung an der Börse. Wie bereits dort erwähnt, ist über diese Funktionen nur wenig bekannt, und wie wir gesehen haben, führt eine falsche Schätzung schnell zu einer deutlich kleineren Verdoppelungsrate.

In diesem Paragraphen wollen wir Ansätze betrachten, die keinerlei Informationen über die Verteilungsfunktionen voraussetzen und als Information für die Wahl eines Portfolios am n-ten Tag höchstens die Wertentwicklungsvektoren $x^{(1)}, x^{(2)}, \ldots, x^{(n-1)}$ der Vortage benutzen.

Wir suchen somit ein kausales Portfolio, d.h. eine Familie b von Portfolios $b^{(i)}(x^1, \ldots, x^{(i-1)})$, deren Wertentwicklungen

$$S_n = \prod_{i=1}^{n} \sum_{k=1}^{m} b_k^{(i)}(x^1, \ldots, x^{(i-1)}) x_k^{(i)}$$

in einem noch zu präzisierenden Sinne „möglichst gut“ sein sollen.

Entsprechende Anlagestrategien werden als *universelle Portfolios* bezeichnet; je nachdem, ob wir von einem festen n ausgehen oder uns eher für die Asymptotik von S_n interessieren, reden wir von einem *universellen Portfolio mit festem Horizont* oder einem *horizontfreien universellen Portfolio.*

Wir wollen uns hier auf den einfachsten Fall beschränken, ein von THOMAS COVER vorgeschlagenes universelles Portfolio mit festem Horizont.

THOMAS M. COVER wurde 1938 im kalifornischen San Bernardino geboren. Er studierte zunächst Physik am Massachusetts Institute of Technology; nach seinem Bachelorabschluß 1960 wechselte er zur Elektrotechnik an die Stanford University, wo er 1961

seinen Master und 1964 seinen PhD bekam. Er blieb in Stanford, zunächst als Assistant Professor der Elektrotechnik, dann als Associate Professor, ab 1971 auch für Statistik; seit 1972 hat er einen Lehrstuhl für Elektrotechnik und Statistik, seit 1994 einen *endowed chair*. Er war unter anderem schon Präsident von IEEE, der internationalen Berufsorganisation der Elektro- und Elektronikingenieure, und beratender Statistiker der kalifornischen Staatslotterie. Die meisten kennen ihn wohl als Autor des Buchs *Elements of Information Theory* (zusammen mit JOY THOMAS), das den ersten beiden Kapiteln dieser Vorlesung zu Grunde liegt.

COVERs Ansatz besteht darin, daß er Investoren betrachtet, die mit einem festen Portfolio $b = (b_1, \ldots, b_m)$ arbeiten. Ein derartiger Investor realisiert eine Wertentwicklung von

$$S_n(b, x) = \prod_{i=1}^{n} \left\langle b, x^{(i)} \right\rangle = \prod_{i=1}^{n} \sum_{k=1}^{m} b_k x_k^{(i)} ,$$

die natürlich stark davon abhängt, wie gut das Portfolio b an den Börsenverlauf $x = (x^{(1)}, \ldots, x^{(n)})$ angepaßt ist. Er möchte sich mit dem erfolgreichsten dieser Investoren vergleichen, also mit einem, der ein Portfolio b^* anwendet, für das $S_n(b^*, x) \geq S_n(b, x)$ ist für alle möglichen Wahlen von b. Ein solches b^* existiert, da die Menge $\mathcal{B}$ aller möglicher Portfolios kompakt ist und $S_n(b, x)$ stetig. Es garantiert zwar keinen Gewinn – bei einem großen Börsencrash wie 1929, 1987 oder 2008 wird man auch mit b^* Verluste machen – aber man kann auch im Verlustfall sicher sein, daß kein anderes konstantes Portfolio einen geringeren Verlust ergeben hätte.

Nun kann allerdings THOMAS COVER genauso wenig in die Zukunft sehen wie der Rest von uns; er weiß also, daß seine Chancen, die nur mit vorheriger Kenntnis aller $x^{(i)}$ realisierbare Strategie b^* zu treffen, äußerst gering sind. Stattdessen versucht er, eine kausale Strategie zu finden, bei der sich die Wertentwicklung möglichst wenig von der des Portfolios b^* unterscheidet.

Diese Wertentwicklung hängt ebenfalls ab von sämtlichen $x^{(i)}$; im Voraus kann man höchstens versuchen, beispielsweise den Erwartungswert des Quotienten $S_n(b, x)/S_n(b^*, x)$ zu optimieren. Dessen Berechnung setzt jedoch voraus, daß wir die Wahrscheinlichkeitsverteilung für die $x^{(i)}$ kennen, und in diesem Paragraphen wollen wir ja ohne deren Kenntnis auskommen. Deshalb wählt COVER ein anderes Kriterium:

Selbst bei der aus Sicht unserer Strategie schlimmstmöglichen Börsenentwicklung soll $S_n(b,x)/S_n(b^*,x)$ möglichst klein sein.

Die Grundidee für seinen Ansatz ist einfach: Falls wir im Voraus wüßten, wie sich der Markt entwickelt, würden wir jeden Morgen das gesamte Anlagekapital in die Aktie investieren, die an jeweiligen Tag die beste Wertsteigerung (oder, bei einem Crash, den geringsten Wertverlust) bringt. Da wir diese Aktie aber erst am Abend kennen, betrachten wir stattdessen *alle* Strategien, die jeden Tag das gesamte vorhandene Kapital auf eine einzige Aktie setzen, für jede der n^m Funktionen

$$j\colon \{1,\dots,n\} \to \{1,\dots,m\}$$

also die Strategie, die am i-ten Tag alles auf die Aktie $j(i)$ setzt; die Menge alle dieser Funktionen j bezeichnen wir mit $\mathcal{J}$.

Für sich allein betrachtet kann jede dieser Strategien sehr riskant sein; mit einem „Portfolio“, in dem *alle* diese Strategien vertreten sind, sollten wir aber in der Lage sein, einen vernünftigen Kompromiss zwischen Wachstum und Risiko zu erreichen.

Wir nehmen also an, daß wir für jede der n^m Funktionen j einen gewissen Anteil $w(j)$ des Ausgangskapitals in die entsprechende Anlagestrategie investieren. Dieser Anteil wird während der n-tägigen Laufzeit multipliziert mit $x^{(1)}_{j(1)} \cdots x^{(n)}_{j(n)}$; das Gesamtkapital wird also multipliziert mit

$$S(w,x) \underset{\text{def}}{=} \sum_{j\in\mathcal{J}} w(j) x^{(1)}_{j(1)} \cdots x^{(n)}_{j(n)} .$$

Das Kapital eines Investors, der auf ein festes Portfolio $b = (b_1,\dots,b_m)$ setzt wird multipliziert mit

$$S(b,x) = \prod_{i=1}^{n} \sum_{k=1}^{m} b_k x^{(i)}_k = \sum_{j\in\mathcal{J}} \prod_{i=1}^{n} \bigl(b_{j(i)} x^{(i)}_{j(i)}\bigr) = \sum_{j\in\mathcal{J}} \prod_{i=1}^{n} b_{j(i)} \prod_{i=1}^{m} x^{(i)}_{j(i)} ;$$

das Verhältnis zwischen den beiden Wertentwicklungen ist also

$$\frac{S(w,x)}{S(b,x)} = \frac{\displaystyle\sum_{j\in\mathcal{J}} w(j) \prod_{i=1}^{m} x^{(i)}_{j(i)}}{\displaystyle\sum_{j\in\mathcal{J}} \prod_{i=1}^{n} b_{j(i)} \prod_{i=1}^{m} x^{(i)}_{j(i)}} .$$

Dieses Verhältnis möchten wir selbst für das beste feste Portfolio b^* möglichst groß werden lassen; wir müssen allerdings realistischerweise davon ausgehen, daß es praktisch immer kleiner als eins sein wird: b^* wird schließlich im Nachhinein berechnet aufgrund der tatsächlichen Wertentwicklung der Aktien, und obwohl wir mit unserem kausalen Portfolio eine größere Flexibilität haben als ein Investor mit einem konstanten Portfolio, wird das *optimale* konstante Portfolio eine Wertsteigerung haben, die wir mit unserer fehlenden Information über die künftige Entwicklung der Aktien nur mit einer verschwindend geringen Wahrscheinlichkeit übertreffen können. Wir reden hier also von der Maximierung einer Größe, die im Allgemeinen deutlich unter eins liegen wird.

Wir könnten uns als eventuell realisierbares Ziel setzen, daß wir Tag für Tag im Durchschnitt wenigstens einen gewissen Prozentsatz der Wertsteigerung des optimalen konstanten Portfolios erreichen können, aber zumindest für große n wäre selbst das ein sehr unbefriedigendes Ergebnis: Auch die Folgen $0{,}9^n$ und $0,99^n$ konvergieren schließlich recht schnell gegen null.

Auch die Strategie von COVER führt auf eine Wertentwicklung, die fast sicher langfristig schlechter sein wird als die mit dem im Nachhinein berechneten optimalen konstanten Portfolio, aber das Verhältnis der Wertentwicklungen geht zumindest langfristig sehr viel langsamer gegen null als jede Folge $(q^n)_{n\in\mathbb{N}}$ für irgendein $q < 1$.

Gemäß der Philosophie von COVER wollen wir auch im schlimmsten Fall noch einen möglichst großen Quotienten haben. Eine Abschätzung für diesen schlimmsten Fall liefert uns das folgende

Lemma: $p_i, q_i \geq 0$ seien reelle Zahlen, und für mindestens ein i sei $(p_i, q_i) \neq (0, 0)$. Dann ist

$$\frac{\sum_{i=1}^{n} p_i}{\sum_{i=1}^{n} q_i} \geq \min_i \frac{p_i}{q_i},$$

wobei für die Minimumsbildung nur Indizes i berücksichtigt werden mit $(p_i, q_i) \neq (0, 0)$.

Beweis: j sei der Index, für den p_j/q_j minimal ist. Falls q_j verschwindet, ist $p_j/q_j = \infty$, und wir müssen nichts beweisen. Falls p_j verschwindet, haben wir die triviale Aussage, daß der Quotient links nicht negativ sein kann. Somit können wir uns beschränken auf den Fall, daß p_j und q_j beide positiv sind. Dann ist

$$\frac{\sum_{i=1}^{n} p_i}{\sum_{i=1}^{n} q_i} = \frac{p_j}{q_j} \frac{\sum_{i=1}^{n} \frac{p_i}{p_j}}{\sum_{i=1}^{n} \frac{q_i}{q_j}} .$$

Nach Wahl von j ist $p_i/q_i \geq p_j/q_j$, also auch $p_i/p_j \geq q_i/q_j$; im zweiten Bruch ist daher der Zähler größer oder gleich dem Nenner, woraus die Behauptung folgt.

■

Auf unsere Situation angewandt, liefert dieses Lemma die Abschätzung

$$\frac{S(w,x)}{S(b,x)} = \frac{\sum_{j\in\mathcal{J}} w(j) \prod_{i=1}^{m} x_{j(i)}^{(i)}}{\sum_{j\in\mathcal{J}} \prod_{i=1}^{n} b_{j(i)} \prod_{i=1}^{m} x_{j(i)}^{(i)}}$$

$$\geq \min_{j\in\mathcal{J}} \frac{w(j) \prod_{i=1}^{m} x_{j(i)}^{(i)}}{\prod_{i=1}^{n} b_{j(i)} \prod_{i=1}^{m} x_{j(i)}^{(i)}} = \min_{j\in\mathcal{J}} \frac{w(j)}{\prod_{i=1}^{n} b_{j(i)}} .$$

Da wir nicht wissen, für welches j das Minimum angenommen wird, müssen wir dafür Sorge tragen, daß die Quotienten, über die wir hier das Minimum bilden, *allesamt* nicht zu klein werden. Dies erreicht COVER dadurch, daß er $w(j)$ proportional zum Maximalwert des Nenners wählt:

$$w(j) \underset{\text{def}}{=} c \max_{\substack{b\in\mathbb{R}_{\geq 0}^{m} \\ b_1+\cdots+b_m=1}} \prod_{i=1}^{n} b_{j(i)} ,$$

wobei die Proportionalitätskonstante natürlich so gewählt werden muß, daß die Summe aller $w(j)$ gleich eins ist.

Bezeichnet $n_k = n_k(j)$ die Anzahl der Tage, an denen $j(i) = k$ ist, können wir den Nenner auch schreiben als

$$\prod_{i=1}^{n} b_{j(i)} = \prod_{k=1}^{m} b_k^{n_k} ;$$

wir müssen also die Funktion $f(b) = \prod_{k=1}^{m} b_k^{n_k}$ maximieren unter der Nebenbedingung $g(b) = \sum_{k=1}^{m} b_k = 1$. Der Gradient von g ist der konstante Vektor mit lauter Einsen; im Maximum müssen daher nach LAGRANGE alle partiellen Ableitungen

$$\frac{\partial f}{\partial b_k} = n_k b_k^{n_k-1} \prod_{\substack{\ell=1 \\ \ell\neq k}}^{m} b_k^{n_k} = \frac{n_k}{b_k} f(b)$$

übereinstimmen, d.h. alle Quotienten n_k/b_k müssen gleich sein. Da die Summe der n_k gleich n, die der $b_k = 1$ ist, folgt $b_k = \frac{n_k}{n}$; der Maximalwert von f ist also

$$f\left(\frac{n_1}{n}, \ldots, \frac{n_m}{n}\right) = \prod_{k=1}^{m} \left(\frac{n_k}{n}\right)^{n_k} = n^{-n} \prod_{k=1}^{n} n_k^{n_k} .$$

Der Logarithmus hiervon ist

$$\sum_{k=1}^{m} n_k \log \frac{n_k}{n} = n \sum_{k=1}^{m} \frac{n_k}{n} \log \frac{n_k}{n} ;$$

wir können den Maximalwert also auch schreiben als

$$2^{-nH\left(\frac{n_1}{n}, \ldots, \frac{n_m}{n}\right)}$$

mit der SHANNONschen Entropiefunktion

$$H(p_1, \ldots . p_m) = -\sum p_k \log_2 p_k .$$

Damit setzen wir also

$$w(j) = c \prod_{k=1}^{m} \left(\frac{n_k(j)}{n}\right)^{n_k(j)} ,$$

und um das auch wirklich berechnen zu können, müssen wir noch die Konstante c bestimmen.

Unter den m^n Funktionen $j\colon \{1,\dots,n\} \to \{1,\dots,m\}$ gibt es

$$\binom{n}{n_1,\dots,n_m} = \frac{n!}{n_1!\cdots n_m!}$$

Funktionen, die n_1-mal den Wert 1, n_2-mal den Wert 2, …, n_m-mal den Wert m annehmen; somit ist

$$1 = \sum_{j\in\mathcal{J}} w(j) = c \sum_{\substack{n_1,\dots,n_m \\ n_1+\cdots+n_m=n}} \frac{n!}{n_1!\cdots n_m!}$$

und

$$c^{-1} = \sum_{\substack{n_1,\dots,n_m \\ n_1+\cdots+n_m=n}} \frac{n!}{n_1!\cdots n_m!} \left(\frac{n_1}{n}\right)^{n_1} \cdots \left(\frac{n_m}{n}\right)^{n_m} ,$$

wobei natürlich alle Summationsindizes $n_j \geq 0$ sein müssen.

Damit können wir die Zahlen $w(j)$ berechnen; tatsächlich arbeiten wir aber natürlich nicht mit m^n verschiedenen Anlagen, von denen wir täglich (fast) jede von Aktie $j(i)$ auf Aktie $j(i+1)$ umschichten; handhabbar wird die Strategie erst, wenn wir wissen, wie wir jeden Tag das *gesamte* vorhandene Kapital auf die m Aktien verteilen.

Der Teil des Anfangskapitals, der nach Strategie $j \in \mathcal{J}$ investiert wird, trägt genau dann am Tag i zum Portfolio für die Aktie k bei, wenn $j(i) = k$ ist. Zu Beginn des i-ten Tages ist der nach dieser Strategie allerdings nicht mehr einfach der Anteil $w(j)$ des Ausgangskapitals, denn der wurde ja an jedem der Vortage multipliziert mit der Wertentwicklung jeder Akte, die j für diesen Tag aussucht. Das Kapital, das zu Beginn des i-ten Tages in Aktie k investiert wird, ist somit das Ausgangskapital mal

$$\sum_{\substack{j\in\mathcal{J} \\ j(i)=k}} w(j) \prod_{\ell=1}^{i-1} x_{j(\ell)}^{(\ell)} .$$

Damit kennen wir den absoluten Betrag des Investments in Aktie k; um das Portfolio für den i-ten Tag zu bekommen, müssen wir noch durch

den gesamten zur Verfügung stehenden Betrag dividieren und erhalten

$$\hat{b}_k^{(i)} = \frac{\displaystyle\sum_{\substack{j\in\mathcal{J}\\ j(i)=k}} w(j)\prod_{\ell=1}^{i-1} x_{j(\ell)}^{(\ell)}}{\displaystyle\sum_{j\in\mathcal{J}} w(j)\prod_{\ell=1}^{i-1} x_{j(\ell)}^{(\ell)}},$$

wobei $w(j)$ aus der obigen Definition übernommen wird.

Dies definiert ein kausales Portfolio $\hat{b}$, denn um das Portfolio für den i-ten Tag zu berechnen, benötigen wir nur Informationen über die Kursentwicklung der Vortage; außerdem wissen wir, daß für jedes, konstante Portfolio b, insbesondere auch für das Portfolio b^*, das sich nach Ablauf der n Tage als das beste herausstellt, das Verhältnis der Wertentwicklungen $S(\hat{b}, x)$ und $S(b^*, x)$ für jeden Börsenverlauf x größer oder gleich c ist.

Was dieses Resultat wert ist, können wir freilich erst beurteilen, wenn wir wissen, wie sich c als Funktion von n und m verhält. Im Buch von COVER und THOMAS wird zitiert, daß sich c asymptotisch verhält wie ein konstantes Vielfaches von $(n+1)^{-(m-1)/2}$; zumindest für hinreichend große Werte von n ist das besser als jede Folge q^n mit $q < 1$.

Speziell für $m = 2$ ist

$$c^{-1} = \sum_{k=0}^{n} \binom{n}{k} \left(\frac{k}{n}\right)^k \left(\frac{n-k}{n}\right)^{n-k}$$

Für Börsen mit (im allgemeinen deutlich) mehr als zwei Aktien sind die hier angegebenen Formeln nicht wirklich praktikabel; außerdem beziehen sie sich nur auf den Fall eines festen Zeithorizonts. Beide Probleme sind lösbar: Im Buch von COVER und THOMAS werden auch universelle Portfolios ohne festen Zeithorizont behandelt, und in verschiedenen Arbeiten von COVER und anderen werden auch Algorithmen aufgestellt, mit denen man universelle Portfolios sowohl bei festem als auch bei unbestimmtem Horizont effizient berechnen kann.

Viele wichtige Arbeiten über KELLYs Wettstrategie, ihre Weiterentwicklungen und auch einige hier nicht behandelte Anwendungen etwa auf Glücksspiele sind gesammelt im Buch

LEONARD C. MACLEAN, WILLIAM T. ZIEMBA, EDWARD O. THORP [HRSG]: The Kelly Capital Growth Investment Criterion: Theory and Practice, *World Scientific,* 2012

Wer also mehr über die inn diesem Kapitel angeschnittene Thematik wissen möchte, finden hier (und auch im Buch von COVER und THOMAS) einen guten Ausgangspunkt für weitere Lektüre.

Kapitel 3
Information erschließen

Im Februar 2011 veröffentlichten MARTIN HILBERT von der University of Southern California und PRISCILA LÓPEZ von der Open University of Catalonia eine Arbeit mit dem Titel *The World's Technological Capacity to Store, Communicate, and Compute Information.* Darin schätzen sie, daß die Menschheit im Jahre 2007 bei optimaler Datenkomprimierung $2{,}9 \cdot 19^{20}$ Byte Information speichern konnte und daß die Summe aller kommunizierten Information in diesem Jahr sogar bei $2 \cdot 10^{21}$ Byte lag. Unter der gespeicherten Information befinden sich natürlich auch viele Musikstücke auf privaten mp3-Playern und Filme auf privaten DVDs, aber auch die allgemein zugängliche Information liegt weit über dem, was ein Einzelner überblicken kann. Spätestens seit der Jahrtausendwende ist allein das World Wide Web so groß geworden, daß keine Suchmaschine mehr seinen Inhalt von einer Redaktion aus menschlichen Spezialisten ordnen lassen kann; benötigt werden Algorithmen, mit denen dies Computer automatisch erledigen können. Da die verfügbare Information zumindest bislang ungefähr im gleichen Tempo anstieg wie die Rechenkraft pro Euro der jeweils aktuellen Computer, hat ein solcher Ansatz Chancen, auch langfristig durchführbar zu bleiben.

Künstliche Intelligenz, Computerlinguistik und ähnliche Forschungsgebiete sind allerdings noch weit davon entfernt, einen Computer allgemeine Texte „verstehen" zu lassen; lediglich bei experimentellen Systemen mit sehr reduziertem Vokabular können Computer ein gewisses Textverständnis simulieren.

Reale Systeme für praktische Anwendungen müssen daher mit ziemlich groben und einfachen Methoden arbeiten; perfekte Ergebnisse kann man

unter diesen Umständen zwar nicht erwarten, aber – wie der Erfolg von Suchmaschinen wie Google zeigt – sind die Resultate doch erstaunlich gut.

§1: Vektorraummodelle

Das erste funktionierende System zur Informationssuche in Textdatenbanken hieß *Smart;* es wurde zwischen 1962 und 1965 unter Leitung von GERARD SALTON an der Harvard University entwickelt. Damals arbeitete es im wesentlichen mit reiner Textsuche; später an der Cornell University entwickelten SALTON und seine Mitarbeiter wesentlich feinere Methoden. Insbesondere verwendeten sie ab Anfang der Siebzigerjahre zunehmend Methoden aus der Linearen Algebra.

Grundlage für den Einsatz entsprechender Algorithmen ist die Term-Dokument-Matrix der Dokumentsammlung: Wir betrachten eine gewisse Menge von Begriffen; bei Fachdatenbanken kann es sich dabei um eine vordefinierte Liste von Stichworten handeln, bei Internetsuchmaschinen aber auch um die Menge aller möglicher Wörter einer Sprache (etwa dreißig Tausend) und eventuell auch noch Falschschreibungen, Eigennamen und so weiter.

Gerade bei Internetsuchmaschinen wird vorher oft auch noch das sogenannte *stemming* praktiziert: Als mögliche Terme gelten nicht die Wörter, sondern die Wortstämme, so daß Flektionsendungen, Verwendung als Substantiv, Adjektiv oder Verb keine Rolle spielen: Beispielsweise werden Information, Informationen, informieren, informierte, informativ *usw.* als *ein* einziger Suchbegriff behandelt.

Manche Suchbegriffe wie Artikel oder häufige Präpositionen sind so unspezifisch, daß sie kaum zum Auffinden geeigneter Dokumente beitragen können; diese werden oft auf eine *Stopliste* gesetzt und zumindest bei Suchanfragen, die auch noch andere Begriffe enthalten, nicht berücksichtigt. (Google findet allerdings auch auf die Anfrage „die“ noch Seiten; ganz unter den Tisch fallen sie also zumindest dort nicht.) Was auf die Stopliste kommt, hängt natürlich von der Art der Anwendung ab; einer der Pioniere der automatischen Textsuche, die Firma Boeing,

erschließt ihren Serviceingenieuren die sämtlichen Handbücher durch eine Suchmaschine, die beispielsweise das Wort „Flugzeug“ auf ihrer Stopliste hat – die Firma stellt schließlich keine Rasenmäher her.

Zur Menge aller verbliebener Begriffe wird üblicherweise zunächst ein sogenannter *invertierter Index* gebildet, d.h. für jeden Begriff wird die Liste aller Dokumente zusammengestellt, in denen er vorkommt, gegebenenfalls mit Zusatzinformationen wo und wie oft. Dieser Index wird von sogenannten *Crawlern* zusammengestellt, die periodisch das *world wide web* nach Dokumenten durchsuchen.

Die Term-Dokument-Matrix hat für jeden möglichen Suchbegriff eine Zeile und für jedes Dokument eine Spalte; der Eintrag in der i-ten Zeile und j-ten Spalte gibt an, wie wichtig der i-te Begriff für das j-te Dokument ist.

Im einfachsten Fall sind alle Einträge entweder 0 oder 1, je nachdem, ob der Begriff vorkommt oder nicht; es gibt aber eine ganze Reihe weiterer Strategien, von denen wir noch einige betrachten werden. Oft werden auch die Spalten auf Länge eins normiert; der Grund dafür wird gleich klar werden. In jedem Fall ist die Matrix *spärlich besetzt,* d.h. nur ein Bruchteil der Einträge ist von null verschieden.

Eine Suchanfrage kann als Vektor betrachtet werden, der für jeden möglichen Suchbegriff einen Eintrag hat; die nicht verschwindenden Einträge geben an, welche Begriffe, gegebenenfalls mit welcher Wichtigkeit, in der Anfrage vorkommen.

Suchanfrage und Dokumente werden also dargestellt durch Vektoren aus ein und demselben Vektorraum; ein Dokument paßt umso besser zur Anfrage, je ähnlicher die beiden Vektoren zueinander sind.

Um die „Ähnlichkeit“ zweier Vektoren in diesem Zusammenhang zu definieren, betrachten wir ein Beispiel:

Wir haben vier Dokumente und die drei Suchbegriffe *Jean, Paul* und *Sartre.* Im ersten Dokument kommen *Jean* und *Paul* je zweimal vor, im zweiten *Jean* zweimal und *Paul* dreimal; von *Sartre* ist in beiden Dokumenten nicht die Rede. Im dritten Dokument kommen alle drei Begriffe je zweimal vor, im vierten schließlich *Jean* und *Paul* je zweimal,

Sartre dreimal. Die Suchanfrage sei *Jean Paul;* wir interessieren uns also für den deutschen Schriftsteller JOHANN PAUL FRIEDRICH RICHTER (1763–1825), der unter dem Pseudonym JEAN PAUL publizierte.

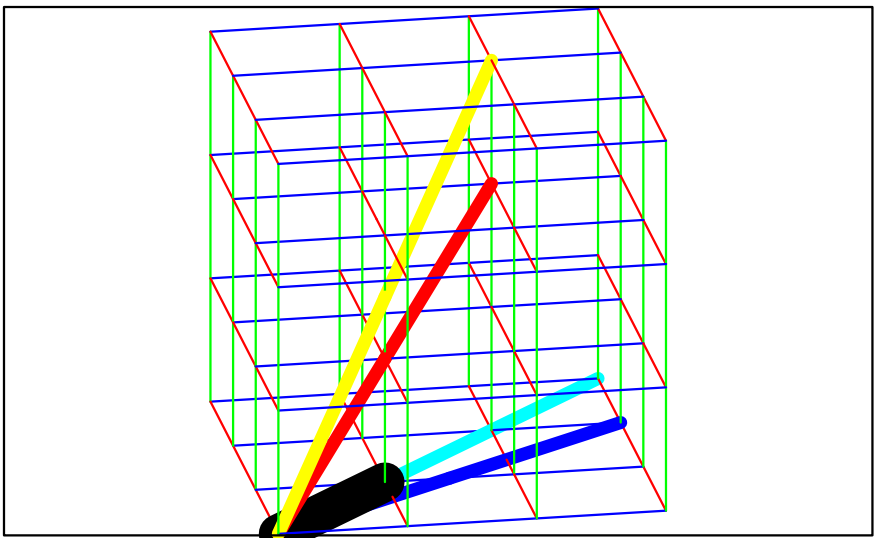
Es ist ziemlich klar, daß wohl nur die ersten beiden Dokumente für diese Anfrage relevant sind; die letzten beiden dürften von dem französischen Philosophen und Schriftsteller JEAN PAUL SARTRE (1905–1980) handeln, nach dem man wohl eher nicht mit der Anfrage *Jean Paul* suchen dürfte.

Die heute gebräuchlichen Suchmaschinen erkennen dies und liefern in erster Linie Dokumente über JEAN PAUL; es gibt aber immer noch online Buchhandlungen (bei meinem Test im Mai 2011 etwa libri.de), die bei der Suche nach JEAN PAUL hauptsächlich Bücher von JEAN PAUL SARTRE finden.

Um zu sehen, wie sich das vermeiden läßt, stellen wir zunächst die Term-Dokument-Matrix auf, wobei wir hier als *Wichtigkeit* einfach die Anzahl der Vorkommen nehmen:

	1	2	3	4
Jean	2	2	2	2
Paul	2	3	2	2
Sartre	0	0	2	3

Die Suchanfrage entspricht dem Spaltenvektor zu $(1, 1, 0)$.



Die Abbildung zeigt die vier Spaltenvektoren der Dokumente in deren jeweiligen Farben sowie den schwarzen Vektor der Suchanfrage. Natürlich sind die Dokumentenvektoren allesamt deutlich länger als der Fragevektor, aber wie man sieht, unterscheiden sich die *Richtungen* der ersten beiden Dokumentenvektoren gar nicht oder kaum von der des Anfragevektors, wohingegen der dritte und der vierte deutlich andere Richtungen haben.

Somit bietet sich als eine einfache Strategie zum Vergleich zwischen Anfrage- und Dokumentenvektoren an, die Berechnung des Winkels an. Da es uns nicht wirklich auf die genauen Werte der Winkel ankommt, sondern nur darauf, ob sie nahe beim Nullwinkel liegen, können wir stattdessen auch die einfacher zu berechnenden Kosinuswerte nehmen und ein Dokument dann als relevant im Sinne der Suchanfrage betrachten, wenn dieser Kosinus hinreichend nahe bei eins liegt. Sofern wir alle Spaltenvektoren der Term-Dokument-Matrix sowie auch den Vektor der Suchanfrage auf Länge eins normieren, können wir diesen Kosinuswert einfach als Skalarprodukt der beiden Vektoren berechnen. Im Falle einer Suchmaschine handelt es sich dabei zwar um Vektoren in einem Vektorraum, dessen Dimension bei rund dreißig Tausend liegen dürfte. Da kaum eine Suchanfrage mehr als drei Terme enthält, müssen wir tatsächlich nur wenige Produkte berechnen und aufaddieren.

§2: Glätten durch orthogonale Projektion

In der Term-Dokument-Matrix ist jedes Dokument repräsentiert durch einen Vektor, der angibt, mit welchem Gewicht welche Terme im Dokument vorkommen. Offensichtlich steckt in der Wahl dieses Vektors viel Willkür, und auch wenn man nach einem einheitlichen Verfahren arbeitet, können inhaltlich sehr ähnlichen Dokumenten recht unterschiedliche Vektoren zugeordnet werden. Hinzu kommt, daß Suchanfragen zwangsläufig zu sehr einfachen Vektoren führen, die in ein sehr viel gröberes Raster passen als die Dokumentenvektoren. Wir können hoffen, daß die Qualität der Suchergebnisse steigt und der Speicherplatzbedarf für die Term-Dokument-Matrix sinkt, wenn wir die Dokumentenvektoren zu Äquivalenzklassen zusammenfassen.

Eine solche Zusammenfassung muß natürlich automatisch erfolgen; alles andere wäre bei wirklich umfangreichen Sammlungen von Dokumenten völlig unrealistisch.

Wir können zumindest informell so tun, als sei der Dokumentenvektor zusammengesetzt aus zwei Komponenten: dem „wirklichen“ Inhalt des Dokuments und einer Art „Rauschen“, das von Zufälligkeiten der Wortwahl und Ähnlichem abhängt. Auch wenn zumindest ich keine Chance sehe, diese beiden Komponenten auch nur einigermaßen präzise zu definieren, befinden wir uns damit doch in einer Situation, mit der wir von anderen Anwendungen her vertraut sind:

a) Lotfußpunkte

Auch hier zerlegen wir einen Vektor in zwei Komponenten: Wenn, im einfachsten Fall, ein Vektor $w \in \mathbb{R}^2$ senkrecht projiziert werden soll auf die Gerade durch den Nullpunkt mit Steigungsvektor u, wollen wir w darstellen als Summe eines Vektors parallel zu u und eines auf u senkrecht stehenden Vektors v:

$$w = \lambda u + v \quad \text{mit} \quad \lambda \in \mathbb{R} \quad \text{und} \quad v \perp u\,.$$

Bilden wir auf beiden Seiten das Skalarprodukt mit u, erhalten wir die Gleichung

$$\langle w, u\rangle = \lambda \langle u, u\rangle + \langle v, u\rangle = \lambda \langle u, u\rangle \quad \text{oder} \quad \lambda = \frac{\langle w, u\rangle}{\langle u, u\rangle}\,.$$

Entsprechend können wir auch im Höherdimensionalen vorgehen: Um einen Vektor $w \in \mathbb{R}^n$ zu projizieren auf den Unterraum, der von den Vektoren $u_1, \ldots, u_r$ aufgespannt wird, zerlegen wir w in eine Linearkombination der u_i sowie einen Lotvektor v, der auf allen u_i senkrecht steht:

$$w = \lambda_1 u_1 + \cdots + \lambda_r u_r + v \quad \text{mit} \quad v \perp u_1, \ldots, v \perp u_r\,.$$

Skalarmultiplikation mit u_i macht daraus

$$\langle w, u_i\rangle = \lambda_1 \langle u_1, u_i\rangle + \cdots + \lambda_r \langle u_r, u_i\rangle\,,$$

und die r so erhaltenen linearen Gleichungen bilden ein lineares Gleichungssystem für die gesuchten Parameter $\lambda_1, \ldots, \lambda_r$.

Besonders einfach wird die Situation, wenn die u_i eine Orthonormalbasis des betrachteten Unterraums bilden, wenn also $\langle u_i, u_j \rangle$ für $i \neq j$ verschwindet und für $i = j$ eins ist. Dann werden die Gleichungen einfach zu $\langle w, u_i \rangle = \lambda_i$. Man beachte, daß wir in keinem Fall den Vektor v wirklich ausrechnen müssen.

b) Überbestimmte lineare Gleichungssysteme

Wenn in einem linearen Gleichungssystem

$$\begin{aligned} a_{11}x_1 + \cdots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + \cdots + a_{2n}x_n &= b_2 \\ \vdots \qquad\qquad & \quad \vdots \\ a_{m1}x_1 + \cdots + a_{mn}x_n &= b_m \end{aligned}$$

die Anzahl m der Gleichungen größer ist als die Anzahl n der Unbekannten, gibt es im allgemeinen keine Lösung. Nun kann es, je nach Anwendung, allerdings vorkommen, daß das Gleichungssystem aus physikalischen oder sonstigen Gründen eine Lösung haben müßte, daß aber die a_{ij} und oder b_i durch Meß- oder Rundungsfehler verfälscht sind und das Gleichungssystem erst dadurch unlösbar wird. Unsere Aufgabe in solchen Fällen besteht darin, eine „Lösung“ $(x_1, \ldots, x_n)$ zu finden derart, daß die Unterschiede zwischen den linken und den rechten Seiten möglichst gering sind.

Fassen wir die Koeffizienten $a_{1j}, \ldots, a_{mj}$ zusammen zu einem Vektor $a_j \in \mathbb{R}^m$ und die rechten Seiten zu einem Vektor $b \in \mathbb{R}^m$, suchen wir also Zahlen $x_1, \ldots, x_n$ derart, daß $x_1a_1 + \cdots + x_na_n$ möglichst nahe bei b liegt.

Das Gleichungssystem ist genau dann exakt lösbar, wenn b im von den Vektoren a_j erzeugten Unterraum U von $\mathbb{R}^m$ liegt. Andernfalls besteht unsere beste Strategie darin, daß wir b senkrecht in diesen Untervektorraum projizieren und das Gleichungssystem mit dem projizierten Vektor c als neuer rechter Seite lösen. Da $v = b - c$ senkrecht auf U steht, ist das wieder äquivalent dazu, daß wir reelle Zahlen x_i suchen,

für die gilt

$$x_1 a_1 + \cdots + x_n a_n + v = b \quad \text{mit} \quad v \perp a_1, \ldots, v \perp a_n \,.$$

Das ist genau das Problem, das wir im vorigen Abschnitt gelöst haben.

Besonders übersichtlich wird die Lösung, wenn wir das Gleichungssystem in Matrixform schreiben als $Ax = b$, wobei A diejenige $m \times n$-Matrix bezeichnet, deren Spalten die Vektoren a_j sind. Da dieses Gleichungssystem unlösbar ist, suchen wir tatsächlich eine Lösung von

$$Ax + v = b \,,$$

wobei v auf allen a_j senkrecht stehen soll; die Skalarprodukte $\langle a_j, v \rangle$ müssen also für $j = 1, \ldots, m$ verschwinden.

Diese Orthogonalitätsbedingung läßt sich ebenfalls kompakter mit Matrizen schreiben: Die transponierte Matrix A^T hat in der j-ten Zeile die Einträge des Vektors a_j stehen; die j-te Komponente von $A^T v$ ist also das Skalarprodukt $\langle a_j, v \rangle$. Somit muß für den obigen Vektor v das Produkt $A^T v$ gleich dem Nullvektor sein. Multiplizieren wir daher die Gleichung $Ax + v = b$ von links mit der Matrix A^T, erhalten wir das neue, lösbare Gleichungssystem

$$(A^T A)x = A^T b \,,$$

dessen Lösungsvektor(en) x für das überbestimmte Gleichungssystem das beste ist (sind), was wir bezüglich der EUKLIDischen Norm bekommen können.

c) Lineare Regression

Hauptanwendung solcher überbestimmter linearer Gleichungssystem ist die Ausgleichsrechnung; das bekannteste Beispiel dazu wiederum sind Ausgleichsgeraden: Hier haben wir zwei Meßgrößen x, y, zwischen denen wir einen Zusammenhang der Form $y = ax + b$ erwarten mit unbekannten Parametern a und b. Für x und y haben wir eine Reihe von Messungen (x_i, y_i) für $i = 1, \ldots, N$, und natürlich wird es im allgemeinen keine zwei reellen Zahlen a, b geben, so daß für alle i gilt $y_i = ax_i + b$.

Die N Gleichungen $y_i = ax_i + b$ bilden für $N > 2$ ein solches überbestimmtes lineares Gleichungssystem aus N Gleichungen für die beiden Unbekannten a und b. (Im Gegensatz zu sonst sind hier also a, b unbekannt, während die x_i, y_i bekannt sind.)

Die Matrix A dieses Gleichungssystem hat zwei Spalten: In der ersten stehen die x_i, in der zweiten stehen lauter Einsen. Der Vektor auf der rechten Seite ist der Vektor y, dessen Komponenten die y_i sind.

A^T hat entsprechend zwei Zeilen, wobei in der ersten die x_i stehen und in der zweiten lauter Einsen. Somit ist

$$A^T A = \begin{pmatrix} \sum_{i=1}^{N} x_i^2 & \sum_{i=1}^{N} x_i \\ \sum_{i=1}^{N} x_i & N \end{pmatrix} \quad \text{und} \quad A^T b = \begin{pmatrix} \sum_{i=1}^{N} x_i y_i \\ \sum_{i=1}^{N} y_i \end{pmatrix} .$$

a und b sind somit Lösungen des linearen Gleichungssystems

$$\sum_{i=1}^{N} x_i^2 \cdot a + \sum_{i=1}^{N} x_i \cdot b = \sum_{i=1}^{N} x_i y_i \quad \text{und} \quad \sum_{i=1}^{N} x_i \cdot a + N \cdot b = \sum_{i=1}^{N} y_i ,$$

die man auch leicht als geschlossene Formel angeben kann.

Entsprechend lassen sich auch im Höherdimensionalen zu vorgegebenen Datenpunkten $(x_{i1}, \ldots, x_{i_m}, y_i) \in \mathbb{R}^{m+1}$ beliebige lineare Regressionsansätze der Form

$$y_i = \sum_{j=1}^{m} a_j f_j(x_{i1}, \ldots, x_{im})$$

mit unbestimmten Koeffizienten a_j auf überbestimmte lineare Gleichungssysteme zurückführen, die nach Multiplikation mit der Transponierten der Matrix des Gleichungssystems ein neues System liefern, dessen Lösungen die nach der Methode der kleinsten Quadrate bestmöglichen Koeffizienten sind. Man beachte, daß der Ansatz *nur* in den a_j linear sein muß; die Funktionen f_j können beliebig gewählt werden.

d) Projektion auf optimale affine Teilräume

Oftmals haben wir in unserem Modell keine expliziten Gleichungen, die eine der Variablen als Funktion der anderen darstellen, sondern

wir suchen einfach eine Relation, die eine Wolke von Datenpunkten möglichst gut beschreibt. Hier wollen wir uns auf den einfachsten Fall einer linearen Relation beschränken; wir suchen also einen affinen Teilraum einer vorgegebenen Dimension, in dessen „Nähe“ die Datenpunkte liegen. Da alle wesentlichen Ideen schon im Eindiemsionalen auftreten, wollen wir zunächst der Fall einer Geraden ausführlich betrachten.

Wir haben also N Punkte $p_1, \ldots, p_N \in \mathbb{R}^n$ und suchen dazu eine im Sinne der kleinsten Quadrate optimale Gerade g. Eine Gerade läßt sich schreiben in der Form

$$g = \left\{ a + tm \;\middle|\; t \in \mathbb{R} \right\},$$

wobei wir annehmen können, daß der Vektor m die Länge eins hat.

Ist $q_i = a + t_i m$ die orthogonale Projektion von p_i auf g, so steht der Differenzvektor $p_i - q_i$ senkrecht auf m, d.h.

$$\langle p_i - a - t_i m, m \rangle = \langle p_i - a_i, m \rangle - t_i = 0\,;$$

somit ist $q_i = a + \langle p_i - a, m \rangle\, m$.

Gesucht ist jene Gerade g, für die die Summe der Abstandsquadrate zu g minimal wird; mit $\|x\| \underset{\text{def}}{=} \sqrt{\langle x, x \rangle}$ soll also

$$\sum_{i=1}^{N} \|p_i - q_i\|^2 = \sum_{i=1}^{N} \|(p_i - a - \langle p_i - a, m \rangle\, m\|^2$$

minimal werden.

Wir überlegen uns zunächst, daß dann das arithmetische Mittel (der Schwerpunkt) der p_i auf g liegen muß: Für

$$v \underset{\text{def}}{=} \frac{1}{N} \sum_{i=1}^{N} (p_i - q_i) \quad \text{und} \quad r_i \underset{\text{def}}{=} p_i - q_i - v$$

ist

$$\sum_{i=1}^{N} \|p_i - q_i\|^2 = \sum_{i=1}^{N} \|v + r_i\|^2 = N \langle v, v \rangle + 2 \left\langle v, \sum_{i=1}^{N} r_i \right\rangle + \sum_{i=1}^{N} \langle r_i, r_i \rangle\ .$$

Dabei ist $\sum_{i=1}^{N} r_i = \sum_{i=1}^{N} (p_i - q_i) - Nv = 0$, also

$$\sum_{i=1}^{N} \|p_i - q_i\|^2 = N \|v\|^2 + \sum_{i=1}^{N} \|r_i\|^2 \ .$$

Geometrisch betrachtet ist $r_i = p_i - (q_i + v)$ der Differenzvektor zwischen p_i und dem Vektor $q_i + v$ auf der Geraden

$$\widetilde{g} = \left\{ a + v + tm \;\middle|\; t \in \mathbb{R} \right\} .$$

Der Abstand von p_i zur Geraden $\widetilde{g}$ ist also höchstens gleich der Länge von r_i, und wäre v nicht der Nullvektor, so wäre die Summe der Abstandsquadrate für $\widetilde{g}$ kleiner als für g. Nach Wahl von g muß somit $v = 0$ sein und

$$s \underset{\text{def}}{=} \frac{1}{N} \sum_{i=1}^{N} p_i = \frac{1}{N} \sum_{i=1}^{N} q_i$$

liegt als Schwerpunkt von Punkten auf der Geraden g selbst auf g.

Wir können daher in der Geradengleichung $a = s$ setzen und müssen nun noch einen Vektor m der Länge eins finden, für den die Summe der Abstandsquadrate zu g minimal wird.

Schreiben wir zur Abkürzung $b_i = p_i - s$, so ist

$$q_i = s + \langle p_i - s, m \rangle \, m = s + \langle b_i, m \rangle \, m \ ,$$

also $p_i - q_i = b_i - \langle b_i, m \rangle \, m$ und

$$\begin{aligned} \sum_{i=1}^{N} \|p_i - q_i\|^2 &= \sum_{i=1}^{N} \|b_i - \langle b_i, m \rangle \, m\|^2 \\ &= \sum_{i=1}^{N} \|b_i\|^2 - 2 \sum_{i=1}^{N} \langle b_i, m \rangle \, \langle b_i, m \rangle + \sum_{i=1}^{N} \|\langle b_i, m \rangle \, m\|^2 \\ &= \sum_{i=1}^{N} \|b_i\|^2 - \sum_{i=1}^{N} \|\langle b_i, m \rangle \, m\|^2 \ . \end{aligned}$$

Da die erste Summe in der zweiten Zeile nicht von m abhängt, muß der gesuchte Vektor m die zweite Summe dort maximal machen.

Bezeichnet B die $N \times N$-Matrix, deren Zeilen die Vektoren b_i sind, so ist Bm der Spaltenvektor mit Einträgen $b_i m$, und das Skalarprodukt dieses Vektors mit sich selbst ist gleich der zu maximierenden Summe.

Identifizieren wir Vektoren mit einspaltigen Matrizen, so ist das Skalarprodukt zweier Vektoren u, v gleich dem Matrixprodukt $u^T \cdot v$, wobei u^T die transponierte Matrix bezeichnet von u bezeichnet, also den zugehörigen Zeilenvektor. Das Skalarprodukt von Bm mit sich selbst ist somit gleich

$$(Bm)^T(Bm) = (m^T B^T)(Bm) = m^T(B^T B)m\,.$$

$B^T B$ ist eine symmetrische reelle $N \times N$-Matrix; wie wir wissen, sind alle ihre Eigenwerte reell, und der $\mathbb{R}^N$ hat eine Basis aus Eigenvektoren von $B^T B$.

Wenn wir in dieser Basis rechnen, wird $B^T B$ zur Diagonalmatrix mit den Eigenwerten $\lambda_1, \ldots, \lambda_N$ als Einträgen, und sind $m_1, \ldots, m_N$ die Komponenten von m bezüglich der Basis aus Eigenvektoren, so ist

$$\begin{aligned} m^T(B^T B)m &= (m_1\ m_2\ \ldots\ m_N) \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_N \end{pmatrix} \begin{pmatrix} m_1 \\ m_2 \\ \vdots \\ m_N \end{pmatrix} \\ &= \sum_{i=1}^{N} \lambda_i m_i^2\,. \end{aligned}$$

Da $m^T(B^T B)m$ als Längenquadrat des Vektors Bm nicht negativ werden kann, sind dabei alle Eigenwerte $\lambda_i \geq 0$.

Damit ist klar, daß der gesuchte Vektor m Eigenvektor der Länge eins zum größten Eigenwert von $B^T B$ sein muß. Falls dieser Eigenwert Vielfachheit eins hat, ist m bis aufs Vorzeichen eindeutig bestimmt, und es gibt nur eine Lösungsgerade; andernfalls sind alle Geraden im affinen Teilraum durch v mit einem Richtungsvektor aus dem Eigenraum Lösungen.

Mit minimalen Veränderungen bei den obigen Argumenten ist nun auch klar, was der optimale r-dimensionale affine Teilraum

$$A = \left\{ a + t_1 m_1 + \cdots + t_r m_r \;\middle|\; t_1, \ldots, t_r \in \mathbb{R} \right\}$$

zu den vorgegebenen Punkten ist: Für a können wir wieder den Schwerpunkt der p_i nehmen, und $m_1, \ldots, m_r$ sind die Eigenvektoren zu den r größten Eigenwerten von $B^T B$.

e) Orthogonalität bei Matrizen

Bei den bisherigen Beispielen wußten wir stets, in welchem Untervektorraum die gesuchte Projektion liegen sollte; im Falle der Term-Dokument-Matrix war bislang nur vage die Rede von einem „Inhalt“, der nie exakt definiert wurde.

Angenommen, wir haben zwei Dokumente, die so ähnlich sind, daß wir ihre Inhalte bezüglich praktisch jeder Suchanfrage als äquivalent betrachten. Angesichts der automatischen und rein formalen Zuordnung von Dokumentenvektoren wird es selbst dann immer wieder vorkommen, daß für die beiden Dokumente verschiedene Vektoren berechnet werden. Wenn wir die Länge der Vektoren auf eins oder eine sonstige Konstante normieren, bedeutet diese Verschiedenheit insbesondere, daß die Vektoren linear unabhängig sind.

Für die gesamte Term-Dokument-Matrix hat dies zur Folge, daß es viel mehr linear unabhängige Spalten gibt als eigentlich notwendig. Von daher bietet sich an, auf einen Raum von Matrizen niedrigeren Ranges zu projizieren; um konkrete Zahlenwerte wollen wir uns im Augenblick noch nicht kümmern, und auch die Tatsache, daß Matrizen eines vorgegebenen Rangs (oder auch höchstens eines vorgegebenen Rangs) nur in den seltensten Fällen einen Untervektorraum bilden, soll uns nicht stören.

Von orthogonalen Projektionen können wir erst reden, wenn wir einen Orthogonalitätsbegriff, d.h. also ein Skalarprodukt haben. Wir wählen dazu die einfachste Möglichkeit: Wir fassen eine $n \times m$-Matrix auf als einen Vektor aus $\mathbb{R}^{nm}$ und nehmen dort das übliche Standardskalarprodukt, d.h.

$$\langle A, B \rangle \underset{\text{def}}{=} \sum_{i=1}^{n} \sum_{j=1}^{m} a_{ij} b_{ij} \; .$$

Wer will, kann das auch kompakter formulieren: Wie stumpfsinniges Nachrechnen zeigt, ist

$$\langle A, B\rangle = \text{Spur}(A^T B)\,,$$

was wir allerdings im folgenden nicht brauchen werden. Die Norm $\|A\| = \sqrt{\langle A, A\rangle}$ zu diesem Skalarprodukt wird, um sie von den vielen anderen möglichen Matrixnormen zu unterscheiden, als FROBENIUS-Norm bezeichnet.

f) Orthonormalbasen im Vektorraum der Matrizen

Da wir den Vektorraum $\mathbb{R}^{m\times n}$ der $m \times n$-Matrizen mit dem Vektorraum $\mathbb{R}^{mn}$ identifizieren, haben wir eine Standardbasis, bestehend aus Matrizen, bei denen genau ein Eintrag gleich eins ist und alle anderen gleich null; wie jede Standardbasis eines $\mathbb{R}^N$ ist das natürlich eine Orthonormalbasis. Wir wollen uns überlegen, daß wir uns auch aus zwei beliebigen Orthonormalbasen von $\mathbb{R}^n$ und $\mathbb{R}^m$ eine Orthonormalbasis von $\mathbb{R}^{n\times m}$ konstruieren können.

Dazu definieren wir zunächst für zwei Vektoren

$$v = \begin{pmatrix} v_1 \\ \vdots \\ v_m \end{pmatrix} \in \mathbb{R}^m \quad \text{und} \quad w = \begin{pmatrix} w_1 \\ \vdots \\ v_n \end{pmatrix} \in \mathbb{R}^n$$

deren *Tensorprodukt* als die Matrix

$$v \otimes w = \begin{pmatrix} v_1w_1 & v_1w_2 & \dots v_1w_n \\ v_2w_1 & v_2w_2 & \dots v_2w_n \\ \vdots & \vdots & \ddots \quad \vdots \\ v_mw_1 & v_mw_2 & \dots v_mw_n \end{pmatrix} \in \mathbb{R}^{m\times n}\,.$$

(Das Zeichen „$\otimes$“ wird in diesem Zusammenhang ausgesprochen als „Tensor“).

Lemma: $(v_1, \dots, v_m)$ und $(w_1, \dots, w_n)$ seien Orthonormalbasen von $\mathbb{R}^n$ *bzw.* $\mathbb{R}^m$. Dann bilden die Vektoren $v_i \otimes w_j$ eine Orthonormalbasis des Vektorraums $\mathbb{R}^{n\times m}$.

Beweis: Wir bezeichnen die Komponenten der Vektoren v_i mit $v_{i\mu}$, die der Vektoren w_j mit $w_{j\nu}$. Dann ist

$$\begin{aligned}\left\langle v_i \otimes w_j, v_k \otimes w_\ell \right\rangle &= \sum_{\mu=1}^{m}\sum_{\nu=1}^{n}(v_{i\mu}w_{j\nu})(v_{k\mu}w_{\ell\nu}) \\ &= \sum_{\mu=1}^{m} v_{i\mu}w_{k\mu}\sum_{\nu=1}^{n} w_{j\nu}v_{\ell\nu} \\ &= \left\langle v_i, v_k \right\rangle \left\langle w_j, w_\ell \right\rangle \ .\end{aligned}$$

Ist $i \neq k$ oder $j \neq \ell$, so verschwindet rechts mindestens einer der beiden Faktoren, also auch das Produkt. Ist aber $i = k$ und $j = \ell$, so sind beide Skalarprodukte rechts gleich eins, also auch das Skalarprodukt links. ∎

§3: Die Singulärwertzerlegung

Unser nächstes Ziel ist es, zu einer gegebenen Matrix $A \in \mathbb{R}^{m\times n}$ Orthonormalbasen $v_1, \ldots v_m$ von $\mathbb{R}^m$ und $w_1, \ldots, w_n$ von $\mathbb{R}^n$ zu finden, derart, daß A in der Orthonormalbasis aus den Matrizen $v_i \otimes w_j$ eine möglichst kurze Basisdarstellung hat. Offensichtlich hat jede der Matrizen $v_i \otimes w_j$ den Rang eins, denn jede ihrer Spalten ist proportional zu w_j, und jede ihrer Zeilen ist proportional zu v_i. Eine Matrix vom Rang r muß daher eine Linearkombination von mindestens r Basismatrizen sein; wir wollen eine Basis finden, in der wir mit genau r auskommen.

Satz: Zu jeder linearen Abbildung

$$\varphi\colon \begin{cases} \mathbb{R}^m \to \mathbb{R}^n \\ \quad v \mapsto Av \end{cases}$$

gibt es Orthonormalbasen $v_1, \ldots v_m$ von $\mathbb{R}^m$ und $u_1, \ldots, u_n$ von $\mathbb{R}^n$ derart, daß in der Abbildungsmatrix Σ von φ bezüglich dieser Basen alle Einträge σ_{ij} mit $i \neq j$ verschwinden und für die Einträge σ_{ii} gilt:

$$\sigma_{11} \geq \sigma_{22} \geq \cdots \geq \sigma_{rr} \ .$$

Für $n = m$ ist Σ also eine Diagonalmatrix, für $n > m$ eine durch Nullspalten und für $m > n$ eine durch Nullzeilen erweiterte Diagonalmatrix.

Den *Beweis* führen wir durch Induktion nach dem Minimum der beiden Zahlen m und n:

Ist dieses Minimum gleich eins, so ist $m = 1$ oder $n = 1$ oder beides.

Im Falle $m = 1$ nehmen wir für $\mathbb{R}^m = \mathbb{R}$ die Orthonormalbasis bestehend aus der Eins. Falls A die Nullmatrix ist, können wir für $\mathbb{R}^n$ eine beliebige Orthonormalbasis wählen; andernfalls nehmen wir als ersten Basisvektor den Vektor $\varphi(1)$ dividiert durch seine Länge und ergänzen ihn zu einer Orthonormalbasis von $\mathbb{R}^n$.

Ist $n = 1$, nehmen wir entsprechend für $\mathbb{R}^n = \mathbb{R}$ die Orthonormalbasis bestehend aus der Eins. In $\mathbb{R}^m$ nehmen wir irgendeine Orthonormalbasis des Kerns und ergänzen sie durch einen weiteren Vektor zu einer Orthonormalbasis von ganz $\mathbb{R}^n$; diesen weiteren Vektor betrachten wir als ersten Basisvektor.

Wenn das Minimum größer als eins ist und A die Nullmatrix, können wir für $\mathbb{R}^n$ und $\mathbb{R}^m$ beliebige Orthonormalbasen wählen und alle $\sigma_i = 0$ setzen.

Für alle anderen Matrizen A betrachten wir in $\mathbb{R}^m$ die Einheitssphäre

$$S = \left\{ v \in \mathbb{R}^m \;\middle|\; |v| = 1 \right\}$$

bestehend aus allen Vektoren der Länge eins, und darauf die Abbildung

$$\psi\colon \begin{cases} S \to \mathbb{R} \\ v \mapsto |\varphi(v)| \end{cases},$$

die jedem Vektor $v \in S$ die Länge des Vektors $\varphi(v) = Av \in \mathbb{R}^n$ zuordnet. Da S kompakt ist, nimmt ψ sein Maximum an; dieses sei σ_1 und werde für den Vektor $v_1 \in S$ angenommen. Da A nicht die Nullmatrix ist, kann σ_1 nicht verschwinden; wir können daher dividieren und setzen

$$u_1 = \frac{\varphi(v_1)}{\sigma_1}.$$

Dann ist $\varphi(v_1) = \sigma_1 u_1$, und u_1 ist wie v_1 ein Vektor der Länge eins.

Nach dem Basisergänzungssatz in Verbindung mit dem Orthogonalisierungsverfahren von GRAM und SCHMIDT können wir dazu weitere

Vektoren $v_2, \ldots, v_n$ und $u_2, \ldots, u_m$ finden derart, daß die Vektoren v_i eine Orthonormalbasis von $\mathbb{R}^n$ bilden und die u_j eine von $\mathbb{R}^m$. Bezüglich dieser beiden Basen habe φ die Abbildungsmatrix A_1.

Da die Spaltenvektoren der Abbildungsmatrix die Koeffizienten der Basisdarstellung der Bildvektoren sind und v_1 aus $\sigma_1 u_1$ abgebildet wird, hat die erste Spalte von A_1 in der ersten Zeile den Eintrag σ_1, und alle anderen Einträge verschwinden. Wir wollen uns überlegen, daß auch in der ersten *Zeile* von A_1 alle anderen Einträge verschwinden müssen.

Wir schreiben

$$A_1 = \begin{pmatrix} \sigma_1 & b_{01} & \cdots & b_{0,m-1} \\ 0 & b_{11} & \cdots & b_{1,m-1} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & b_{n-1,1} & \cdots & b_{n-1,m-1} \end{pmatrix}$$

und betrachten den Vektor

$$v = \sigma_1 v_1 + b_{01} v_2 + \cdots + b_{0,m-1} v_m \in \mathbb{R}^m \, .$$

Da er bezüglich einer Orthonormalbasis dargestellt ist, können wir das Quadrat seiner Länge einfach als Summe der Koeffizientenquadrate berechnen, d.h.

$$\|v\|^2 = \sigma_1^2 + \sum_{j=1}^{m-1} b_{0j}^2 \, .$$

Sein Bild unter φ ist

$$\begin{aligned} \varphi(v) &= \sigma_1 \varphi(v_1) + b_{01} \varphi(v_2) + \cdots + b_{0,m-1} \varphi(v_{m-1}) \\ &= \sigma_1^2 u_1 + b_{01} b^{(1)} + \cdots b_{0,m-1} b^{(m-1)} \, , \end{aligned}$$

wobei $b^{(j)}$ den $(j-1)$-ten Spaltenvektor von A_1 bezeichnet. In Koordinaten ausgedrückt ist somit

$$\varphi(v) = \begin{pmatrix} \sigma_1^2 + b_{01}^2 + \cdots + b_{0,m-1}^2 \\ b_{01} b_{11} + \cdots + b_{0,m-1} b_{1,m-1} \\ \vdots \\ b_{01} b_{n-1,1} + \cdots + b_{0,m-1} b_{n-1,m-1} \end{pmatrix} \, .$$

Das Längenquadrat dieses Vektors ist Quadratsumme der Einträge, d.h.

$$\|\varphi(v)\|^2 \geq \left(\sigma_1^2 + b_{01}^2 + \cdots + b_{0,m-1}^2\right)^2 .$$

Der Einheitsvektor

$$v_0 \underset{\text{def}}{=} \frac{v}{\|v\|} = \frac{v}{\sqrt{\sigma_1^2 + b_{01}^2 + \cdots + b_{0,m-1}^2}}$$

wird dementsprechend abgebildet auf den Vektor $\varphi(v)/\|v\|$, dessen Länge größer oder gleich

$$\frac{\sigma_1^2 + b_{01}^2 + \cdots + b_{0,m-1}^2}{\sqrt{\sigma_1^2 + b_{01}^2 + \cdots + b_{0,m-1}^2}} \geq \sqrt{\sigma_1^2 + b_{01}^2 + \cdots + b_{0,m-1}^2}$$

ist. Diese Länge kann aber höchstens gleich σ_1 sein, denn nach Konstruktion ist das ja die größtmögliche Länge für das Bild eines Vektors der Länge eins. Somit müssen alle b_{0j} verschwinden; die Matrix A_1 hat also die Form

$$A_1 = \begin{pmatrix} \sigma_1 & 0 \\ 0 & B \end{pmatrix}$$

mit einer $(n-1) \times (m-1)$-Matrix B. Dies zeigt, daß φ den von v_2 bis v_m erzeugten Untervektorraum des $\mathbb{R}^m$ auf den von u_2 bis u_n erzeugten Untervektorraum des $\mathbb{R}^n$ abbildet. Schränken wir die Abbildung φ ein auf diese beiden Untervektorräume, haben wir jeweils um eins kleinere Dimensionen; nach Induktionsannahme gibt es also Orthonormalbasen dieser Untervektorräume, bezüglich derer die Einschränkung von φ Diagonalgestalt hat. Nach Wahl von σ_1 ist klar, daß alle Diagolaneinträge kleiner oder gleich σ_1 sein müssen.

Ersetzen wir $v_2, \ldots, v_m$ und $u_2, \ldots, u_n$ durch die Vektoren dieser Basen, erhalten wir zusammen mit v_1 und u_1 Orthonormalbasen von $\mathbb{R}^m$ und $\mathbb{R}^n$, bezüglich derer die Abbildungsmatrix Σ von φ keine Einträge σ_{ij} mit $i \neq j$ hat und für die $\sigma_{ii} = \sigma_i$ gilt $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r$, wobei r das Minimum der beiden Dimensionen m und n bezeichnet. ∎

Der Wechsel von der Standardbasis zu dieser Orthonormalbasis wird jeweils durch eine orthogonale Matrix beschrieben; somit haben wir bewiesen:

Satz: Jede reelle $m \times n$-Matrix A läßt sich als ein Produkt $A = U\Sigma V^T$ schreiben mit orthogonalen Matrizen $U \in \mathbb{R}^{m\times m}$ und $V \in \mathbb{R}^{n\times n}$ sowie einer Matrix $\Sigma \in \mathbb{R}^{m\times n}$, in der alle Einträge σ_{ij} mit $i \neq j$ verschwinden und für die restlichen Einträge gilt $\sigma_{11} \geq \sigma_{22} \geq \cdots \geq \sigma_{rr} \geq 0$. ∎

Definition: Die Zahlen $\sigma_i \underset{\text{def}}{=} \sigma_{ii}$ heißen *singuläre Werte* von A; die Spaltenvektoren von U und von V bezeichnen wir als *singuläre Vektoren* von A. Die Zerlegung $A = U\Sigma V^T$ heißt *Singulärwertzerlegung* der Matrix A.

Da die Matrizen U und V orthogonal sind, sind ihre inversen Matrizen einfach die transponierten. Dies können wir ausnützen um die singulären Werte und Vektoren mit bekannten Größen in Verbindung zu bringen:

$$AA^T = U\Sigma V^T V\Sigma^T U^T = U\Sigma\Sigma^T U^T = U\Delta U^T = U\Delta U^{-1}\,,$$

wobei Δ eine Diagonalmatrix mit Einträgen σ_i^2 ist. Somit sind die σ_i die Wurzeln der Eigenwerte von AA^T.

Multiplizieren wir die Gleichung $A = U\Sigma V^T$ von rechts mit V, erhalten wir die Gleichung $AV = U\Sigma$, denn für eine orthogonale Matrix V ist $V^T V = VV^T$ gleich der Einheitsmatrix. Für den i-ten Spaltenvektor v_i von V ist daher $Av_i = \sigma_i u_i$.

Entsprechend können wir $A^T = V\Sigma^T U^T$ von rechts mit U multiplizieren und erhalten $A^T U = V\Sigma^T$; für den i-ten Spaltenvektor u_i von U ist daher $A^T u_i = \sigma_i v_i$.

Fassen wir beides zusammen, erhalten wir die Gleichungen

$$A^T Av_i = \sigma_i^2 v_i \quad \text{und} \quad AA^T u_i = \sigma_i^2 u_i\,;$$

die singulären Vektoren sind also die Eigenvektoren von $A^T A$ *bzw.* AA^T.

Als orthogonale Matrizen sind U und V insbesondere invertierbar; der Rang der Ausgangsmatrix A ist daher gleich dem der Matrix Σ, d.h. gleich der Anzahl r der nicht verschwindenden Singulärwerte.

Da die Einträge der Matrix Σ höchstens dann von Null verschieden sein können, wenn der Zeilenindex gleich dem Spaltenindex ist, wird im

Produkt $U\Sigma$ der i-te Spaltenvektor von U mit σ_i multipliziert. Entsprechend wird im Produkt ΣV^T der i-te Zeilenvektor von V^T, d.h. also die i-te Spalte von V, mit σ_i multipliziert. Für $i > r$ ist $\sigma_i = 0$, die entsprechenden Spalten von U und V spielen also für die Berechnung von $A = U\Sigma V^T$ keinerlei Rolle und müssen somit auch nicht abgespeichert werden. Bezeichnet U_1 die $n \times r$-Matrix aus den ersten r Spaltenvektoren von U, V_1 die $m \times r$-Matrix aus den ersten r Spaltenvektoren von V und Σ_1 die $r \times r$-Diagonalmatrix mit Einträgen $\sigma_1, \ldots . \sigma_r$, so ist also auch

$$A = U_1 \Sigma_1 V_1^T \quad \text{mit} \quad U_1 \in \mathbb{R}^{n \times r}, \quad V_1 \in \mathbb{R}^{m \times r} \quad \text{und} \quad \Sigma_1 \in \mathbb{R}^{r \times r} .$$

Ausgedrückt durch die Spaltenvektoren u_i, v_i von U_1 und V_1 können wir dies auch schreiben als

$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^T = \sum_{i=1}^{r} \sigma_i u_i \otimes v_i$$

mit dem in §2*f)* eingeführten Tensorprodukt.

Die Projektion auf den Raum aller Matrizen vom Rang höchstens s für ein $s \leq r$ liefert uns der folgende

Satz: A sei eine Matrix vom Rang r; ihre Singulärwertzerlegungsei

$$A = U\Sigma V^T = \sum_{i=1}^{r} \sigma_i u_i \otimes v_i .$$

Dann ist für jedes $s \leq r$ die Matrix

$$A_s = \sum_{i=1}^{s} \sigma_i u_i \otimes v_i$$

eine orthogonale Projektion von A auf einen Vektorraum von Matrizen mit Rang höchstens s, und für jede Matrix B vom Rank höchstens s gilt: $\|A - B\| \geq \|A_s - A\|$.

(Als Matrix können wir A_s wie folgt definieren: Die Matrix Σ_s entstehe aus Σ dadurch, daß alle σ_{ii} mit $i > s$ auf Null gesetzt werden. Dann ist $A_s = U\Sigma_s V^T$. Die Matrix A_s ist genau dann eindeutig bestimmt, wenn $\sigma_s > \sigma_{s+1}$ ist; andernfalls gibt es mehrere Lösungen.)

Beweis: Da die sämtlichen u_i *bzw.* v_j jeweils eine Orthonormalbasis des zu Grunde liegenden Vektorraums bilden, bilden die $u_i \otimes v_j$ nach §2*f)* eine Orthonormalbasis des entsprechenden Vektorraums von Matrizen. Wir betrachten eine beliebige Matrix B aus diesem Raum und schreiben sie als

$$B = \sum_{i=1}^{n} \sum_{j=1}^{m} b_{ij} u_i \otimes v_j \; .$$

Da das Quadrat der EUKLIDischen Norm bezüglich einer Orthonormalbasis einfach als Summe der Koeffizientenquadrate berechnet werden kann, ist

$$\|B - A\|^2 = \sum_{i=1}^{\min(m,n)} (b_{ii} - \sigma_i)^2 + \sum_{i \neq j} b_{ij}^2 \; .$$

Wenn dies minimal werden soll, müssen also zunächst alle b_{ij} mit $i \neq j$ verschwinden.

Von den Koeffizienten b_{ii} können für eine Matrix B vom Rang höchstens $s < r \leq \min(n, m)$ nicht mehr als s von Null verschieden sein; wir können daher nicht alle r Koeffizienten $b_{ii} = \sigma_i$ setzen, sondern müssen einige auch auf null setzen. Ein solcher Koeffizient liefert dann einen Beitrag von σ_i^2 zur obigen Summe. Da die σ_i der Größe nach geordnet sind, setzen wir somit $b_{ii} = \sigma_s$ für $i \leq s$ und $b_{ii} = 0$ sonst. ∎

§4: Latente semantische Analyse

Wir haben orthogonale Projektionen und die Singulärwertzerlegungbetrachtet, um damit die Term-Dokument-Matrix zu „entrauschen“; wir wollen also den Vektor v zu einem Dokument auffassen als Summe zweier Vektoren, von denen der eine den „wirklichen“ Inhalt des Dokuments beschreibt, während im anderen all die Zufälligkeiten stecken, die individuelle Wortwahl (Karotte/Möhre, Auto/PKW) und Stil mit sich bringen. Wenn wir diese Zerlegung wirklich definieren und auch durchführen könnten, hätte die Matrix der „Inhaltsvektoren“ sicherlich einen kleineren Rang als die Term-Dokument-Matrix.

Ansatzpunkt der latenten semantischen Analyse ist die logisch unzulässige, aber praktisch bewährte Umkehrung dieser Aussage: Wenn wir die Term-Dokument-Matrix durch eine „benachbarte“ Matrix niedrigeren Rangs ersetzen, steht zu hoffen, daß deren Spalten eher den „Inhaltsvektoren“ entsprechen als die Spalten der Term-Dokument-Matrix.

Ein ähnliches Problem hat auch die numerische Mathematik beim Rechnen mit Gleitkomma-Matrizen: Falls die Spalten a_i einer Matrix einer linearen Gleichung $\sum \lambda_i a_i = 0$ genügen sollten, wird durch allfällige Rundungsfehler die rechte Seite tatsächlich oftmals verschieden vom Nullvektor; der Rang der Gleitkomma-Matrix wird also größer als der Rang der exakten Matrix.

Betrachtet man die singulären Werte einer solchen Matrix, so werden diese meist hinter einem festen Index plötzlich sehr viel kleiner. Diesen Index bezeichnet man als den (nicht wirklich exakt definierten) *numerischen Rang* der Matrix.

Als Beispiel betrachten wir die Matrix

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} \quad \text{mit} \quad AA^T = \begin{pmatrix} 14 & 32 & 50 \\ 32 & 77 & 122 \\ 50 & 122 & 194 \end{pmatrix}$$

Die Eigenwerte von AA^T sind 0 und $\frac{1}{2}(285 \pm 3\sqrt{8881})$, und tatsächlich hat A natürlich nur den Rang zwei.

Ein Programm wie MatLab berechnet jedoch auch zu A eine „inverse“ Matrix (wenn auch mit Warnung über die schlechte Konditionszahl) und kommt auf die singulären Werte 16,85, 1,07 und $4{,}42 \cdot 10^{-16}$. Hier ist der numerische Rang offensichtlich gleich dem Rang zwei der exakten Matrix.

So extrem wie in diesem Beispiel ist der Abfall der singulären Werte im Falle von Term-Dokument-Matrizen natürlich nur selten; trotzdem ist meist *ungefähr* klar, ab wann sie klein genug werden, um vernachlässigt zu werden. Der wohl populärste Ansatz zur latenten semantischen Analyse besteht daher darin, die Term-Dokument-Matrix so auf eine Matrix niedrigeren Rangs zu projizieren und mit dieser zu arbeiten.

Als Beispiel für eine latente semantische Analyse betrachtet

LARS ELDÉN: Matrix Methods in Data Mining and Pattern Recognition, SIAM, 2007

fünf Dokumente folgenden Inhalts:

1. The Google^{TM} matrix P is a model of the internet.
2. P_{ij} is nonzero, if there is a link from Web page j to i.
3. The Google matrix is used to rank all Web pages.
4. The ranking is done by solving a matrix eigenvalue problem.
5. England dropped out of the top ten in the FIFA ranking.

Wenn wir die zehn Suchbegriffe *eigenvalue, England, FIFA, Google, internet, link, matrix, page, rank, Web* zulassen, erhalten wir die Term-Dokumentmatrix

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

Die Suchanfrage „Ranking of Web Pages“ entspricht dem Spaltenvektor zu $(0,0,0,0,0,0,0,1,1,1)$; berechnen wir den Kosinus seines Winkels mit den fünf Spaltenvektoren von A erhalten wir die Werte $0, \frac{2}{3}, \frac{3}{\sqrt{15}} \approx 0{,}775$ und für die beiden letzten Spalten jeweils $\frac{1}{3}$. Demnach wäre das dritte Dokument das passendste, gefolgt vom zweiten, danach gleichrangig das vierte und das fünfte und am unpassendsten das erste.

Tatsächlich ist klar, daß für diese Anfrage das letzte Dokument völlig irrelevant ist, während das erste trotz disjunkter Suchbegriffe durchaus eine gewisse Bedeutung hat.

Die Singulärwertzerlegungvon A ist $A = U_1 \Sigma V^T$ mit

$$U_1 = \begin{pmatrix} -0{,}142 & -0{,}243 & 0 & -0{,}578 & 0{,}364 \\ -0{,}0787 & -0{,}261 & -0{,}385 & 0{,}392 & 0{,}168 \\ -0{,}0787 & -0{,}261 & -0{,}385 & 0{,}392 & 0{,}168 \\ -0{,}392 & 0{,}0274 & 0{,}385 & 0{,}399 & -0{,}251 \\ -0{,}130 & -0{,}0740 & 0{,}385 & 0{,}375 & 0{,}495 \\ -0{,}102 & 0{,}373 & -0{,}192 & -0{,}0346 & 0{,}654 \\ -0{,}535 & -0{,}216 & 0{,}385 & -0{,}179 & 0{,}113 \\ -0{,}365 & 0{,}475 & -0{,}192 & -0{,}0102 & -0{,}0917 \\ -0{,}484 & -0{,}402 & -0{,}385 & -0{,}161 & -0{,}215 \\ -0{,}365 & 0{,}475 & -0{,}192 & -0{,}0102 & -0{,}0917 \end{pmatrix},$$

$$\Sigma = \begin{pmatrix} 2{,}85 & 0 & 0 & 0 & 0 \\ 0 & 1{,}88 & 0 & 0 & 0 \\ 0 & 0 & 1{,}73 & 0 & 0 \\ 0 & 0 & 0 & 1{,}26 & 0 \\ 0 & 0 & 0 & 0 & 0{,}848 \end{pmatrix}$$

und

$$V = \begin{pmatrix} -0{,}370 & -0{,}139 & 0{,}667 & 0{,}472 & 0{,}420 \\ -0{,}291 & 0{,}703 & -0{,}333 & -0{,}0436 & 0{,}555 \\ -0{,}750 & 0{,}191 & 0 & 0{,}0308 & -0{,}633 \\ -0{,}407 & -0{,}457 & 0 & -0{,}728 & 0{,}309 \\ -0{,}225 & -0{,}491 & -0{,}667 & 0{,}494 & 0{,}142 \end{pmatrix}.$$

Setzen wir zur latenten semantischen Analyse die letzten drei dieser Diagonaleinträge auf Null, erhalten wir die neue Matrix

$$\begin{pmatrix} -0{,}142 & -0{,}243 \\ -0{,}0787 & -0{,}261 \\ -0{,}0787 & -0{,}261 \\ -0{,}392 & 0{,}0274 \\ -0{,}130 & -0{,}0740 \\ -0{,}102 & 0{,}373 \\ -0{,}535 & -0{,}216 \\ -0{,}365 & 0{,}475 \\ -0{,}484 & -0{,}402 \\ -0{,}365 & 0{,}475 \end{pmatrix} \begin{pmatrix} 2{,}85 & 0 \\ 0 & 1{,}88 \end{pmatrix} \begin{pmatrix} -0{,}370 & -0{,}291 \\ -0{,}139 & 0{,}703 \\ 0{,}667 & -0{,}333 \\ 0{,}472 & 0{,}044 \\ 0{,}420 & 0{,}555 \end{pmatrix}$$

$$= \begin{pmatrix} 0{,}214 & -0{,}203 & 0{,}218 & 0{,}375 & 0{,}316 \\ 0{,}152 & -0{,}280 & 0{,}0748 & 0{,}316 & 0{,}291 \\ 0{,}152 & -0{,}280 & 0{,}0748 & 0{,}316 & 0{,}291 \\ 0{,}407 & 0{,}362 & 0{,}850 & 0{,}432 & 0{,}226 \\ 0{,}156 & 0{,}00992 & 0{,}251 & 0{,}214 & 0{,}152 \\ 0{,}00992 & 0{,}579 & 0{,}353 & -0{,}203 & -0{,}280 \\ 0{,}622 & 0{,}159 & 1.07 & 0{,}807 & 0{,}542 \\ 0{,}261 & 0{,}932 & 0{,}951 & 0{,}0147 & -0{,}205 \\ 0{,}617 & -0{,}130 & 0{,}891 & 0{,}908 & 0{,}682 \\ 0{,}261 & 0{,}932 & 0{,}951 & 0{,}0147 & -0{,}205 \end{pmatrix}.$$

Berechnen wir nun die Kosinuswerte der Winkel zwischen den Spalten und den Suchanfragen, erhalten wir die neuen Werte

$$0.604, \quad 0.640, \quad 0.743, \quad 0.374 \quad \text{und} \quad 0.140\,,$$

wonach zwar weiterhin das dritte Dokument die beste Antwort ist, allerdings liegen nun sowohl das erste als auch das vierte deutlich vor dem irrelevanten fünften. Der Grund liegt natürlich darin, daß die Projektionen der entsprechenden Spaltenvektoren von A auf den von den ersten beiden Spaltenvektoren von U_1 aufgespannten Untervektorraum des $\mathbb{R}^{10}$ weitaus besser übereinstimmen als die Originale im $\mathbb{R}^{10}$.

§5: Der PageRank von Google

Google begann als studentisches Forschungsprojekt der beiden Doktoranden SERGEY BRIN und LAWRENCE PAGE an der Stanford University im kalifornischen Palo Alto; seine ersten Vorläufer waren auch nur dort auf dem Campus zugänglich. Als Stanfords Präsident JOHN HENNESSY, ein technischer Informatiker, Mitte der neunziger Jahre erstmals von der neuen Suchmaschine hörte, tippte er seinen Namen ein und erhielt gleich als erstes eine Seite von Stanford. Das war ihm bei der damals führenden Suchmaschine AltaVista noch nie passiert und trug sicherlich mit dazu bei, daß Stanford später die Kommerzialisierung von Google nach Kräften förderte.

Die ersten Prototypen berücksichtigten zur Anordnung der Suchergebnisse selbstverständlich noch nicht die heute üblichen über zweihundert

„Signale“; damals gab es im wesentlichen nur ein Kriterium, den *PageRank*. Er ist benannt nach LAWRENCE PAGE und patentiert als US Patent 6 285 999 vom 4. September 2001 mit Anschlußpatent 7 058 628 vom 6. Juni 2006*) Inhaber der Patente ist die Stanford University; als Erfinder ist jeweils LAWRENCE PAGE angegeben.

Im Gegensatz zu den meisten anderen Kriterien ist der PageRank unabhängig von jeder Suchanfrage: Durch ihn sollen *alle* (dem System bekannten) Webseiten nach ihrer Wichtigkeit geordnet werden. Getreu der allgemeinen Philosophie von Google muß diese Wichtigkeit nach einem gut skalierbaren Verfahren ohne menschliche Intervention berechenbar sein.

Die Grundidee dazu ist einfach und auch nicht neu: Seit langem wird immer wieder versucht, die Wichtigkeit wissenschaftlicher Arbeiten rein mechanisch zu bestimmen. Ein einfacher und deshalb gerne verwendeter Ansatz besteht darin, eine Arbeiten nach der Anzahl jener anderer Arbeiten zu beurteilen, in denen sie zitiert wird. Mit Hilfe des *Science Citation Index* und inzwischen auch *CiteSeer* (`citeseer.ist.psu.edu`) und ähnlichen Datenbanken läßt sich diese Anzahl leicht feststellen (oder zumindest schätzen, denn sie hängt natürlich ab vom Umfang der verwendeten Datenbank), und man erhält ein objektives Maß.

Weniger klar ist, was durch dieses Maß gemessen wird, denn an der Spitze stehen praktisch nie Arbeiten, die ein Fachwissenschaftler der entsprechenden Disziplin zu den wichtigsten aus dem betreffenden Zeitraum rechnen würde. Der Grund ist ziemlich klar: Ein kleines Licht mit einer großen Schar mittelmäßiger Schüler, die allesamt ständig den großen Meister zitieren, schneidet hier besser ab als ein Autor, der nur von wenigen hochkarätigen Spezialisten zitiert wird.

Ähnlich sieht es aus, wenn man diese Vorgehensweise auf das *World Wide Web* überträgt. Da eine Volltextsuchmaschine ohnehin Kopien aller ihr bekannter Webseiten im Speicher hat, kann sie zu jeder dieser Seiten

*) US-Patente sind auf dem offiziellen Server `patft.uspto.gov` des *United States Patent and Trademark Office* in HTML zu finden, pdf-Dateien bei `www.pat2pdf.org`.

leicht ermitteln, wie viele andere Seiten darauf verweisen und kann dann als Wichtigkeitsmaß für eine Seite S definieren

$$w_0(S) = \text{Anzahl der Seiten, die auf } S \text{ verweisen .}$$

Die Probleme mit diesem Maß sind im wesentliche dieselben wie oben:

1. Ein Verweis von einer wichtigen Seite sagt mehr aus, als ein Verweis von einer unwichtigen.
2. Wenn eine wichtige Seite auf hundert andere Seiten verweist, kann man das nicht vergleichen mit einem Verweis auf nur eine einzige Seite.
3. Durch Massenproduktion inhaltsleerer Seiten, deren einziger Zweck der Verweis auf eine zu pushende Webseite ist, läßt sich das Maß leicht manipulieren.

PAGE benutzt trotzdem die Informationen, die in der Verweisstruktur des World Wide Web steckt, allerdings mit einer Modifikation, die den ersten beiden Problemen entgegenwirkt und damit zumindest teilweise auch das dritte löst: Eine Seite ist wichtig, wenn wichtige Seiten auf sie verweisen, insbesondere dann, wenn diese nur auf wenige andere Seiten verweisen.

Ein erster Ansatz, dies in eine mathematische Formel umzusetzen, könnte folgender sein: Jede Seite S erhält eine Wichtigkeit $w(S)$, für die folgendes gilt: Sind $R_1, \ldots, R_n$ die Seiten, die auf S verweisen und verweist R_i auf m_i Seiten, so ist

$$w(S) = \sum_{i=1}^{n} \frac{w(R_i)}{m_i} \,. \qquad (*)$$

Dies ist sicherlich eine sinnvolle Forderung, jedoch ist *a priori* nicht klar, ob dadurch eine eindeutige Rangordnung definiert wird: Erst einmal muß untersucht werden, ob es überhaupt eine von der Nullfunktion verschiedene Lösungsfunktion w gibt, danach stellt sich noch das Problem der Eindeutigkeit.

Diese Frage läßt sich einfach beantworten, denn die obige Formel definiert offensichtlich ein lineares Gleichungssystem für die Unbekannten $w(S)$. Um es in eine üblichere Form zu bringen, bezeichnen

wir die der Suchmaschine bekannten Dokumente mit $S_1, \ldots, S_N$, die Anzahl der von Seite S_j ausgehenden Verweise mit m_j und setzen

$$a_{ij} = \begin{cases} \frac{1}{m_j} & \text{falls es einen Verweis } S_j \to S_i \text{ mit } j \neq i \text{ gibt} \\ 0 & \text{sonst} \end{cases} .$$

Dann wird die obige Gleichung zu

$$w_i = \sum_{j=1}^{N} a_{ij} w_j \quad \text{für } i = 1, \ldots, N .$$

Bringen wir hier noch w_i auf die andere Seite, haben wir die Standardform eines homogenen linearen Gleichungssystems:

$$\sum_{j=1}^{n} b_{ij} w_j = 0 \quad \text{mit} \quad b_{ij} = \begin{cases} a_{ij} & \text{falls } i \neq j \\ -1 & \text{falls } i = j \end{cases} .$$

Ein solches System hat immer den Nullvektor als Lösung; weitere Lösungen gibt es genau dann, wenn die Gleichungen linear abhängig sind.

Für eine Seite S_j, von der mindestens ein Verweis ausgeht, ist

$$\sum_{i=1}^{N} a_{ij} = m_j \cdot \frac{1}{m_j} = 1 \quad \text{und damit} \quad \sum_{i=1}^{N} b_{ij} = 0 ;$$

falls von jeder Seite mindestens ein Verweis ausgeht, ist also die Summe aller N linker Seiten gleich Null. In diesem Fall sind daher die Gleichungen linear abhängig und es gibt auch nichttriviale Lösungen.

Nun gibt es allerdings viele Seiten, die auf keine anderen Seiten verweisen, zum Beispiel die pdf-Datei mit dem Text dieses Skriptums. Um trotzdem die Existenz nichttrivialer Lösungen zu garantieren, werden vor allem zwei Strategien angewandt:

1. Man ignoriert zunächst alle Seiten, die auf keine anderen verweisen. Für den Rest löst man das lineare Gleichungssystem und setzt die Lösung dann mittels der Formel $(*)$ fort auf die restlichen Seiten. Da diese Seiten auf nichts verweisen, können sie nie auf der rechten Seite von $(*)$ auftreten; rechts stehen immer nur bereits aus dem ersten Schritt bekannte Gewichte.

2. Man behandelt diese Seiten so, als würden sie auf *jede* andere Seite verweisen, setzt also für eine solche Seite S_j den Wert von a_{ij} für jedes i auf $1/N$. Dann ist auch für solche j die Summe aller a_{ij} gleich eins, so daß obiges Argument die Existenz einer nichttrivialen Lösung zeigt.

Ein weiteres Problem sind Verweise auf nicht (mehr) existente oder einfach nur unzugängliche Seiten, z.B. solche mit Paßwortschutz oder Gebühr. Diese muß die Suchmaschine entweder ignorieren oder aber wie eine Seite ohne ausgehende Verweise behandeln.

Nachdem die Existenz nichttrivialer Lösungen geklärt ist, stellt sich als nächstes die Frage der Eindeutigkeit. Natürlich erfüllen mit jedem Lösungsvektor auch dessen sämtliche Vielfachen das Gleichungssystem; sofern es aber eine Lösung mit ausschließlich nichtnegativen Wichtigkeiten gibt, führen alle positiven Vielfachen davon auf dieselbe Rangordnung. Wir müssen also sicherstellen, daß der Lösungsraum erstens eindimensional ist und zweitens Vektoren ohne negative Komponenten enthält.

Leider kann die Dimension des Lösungsraums deutlich größer sein als eins: Wenn wir die Webseiten $S_1, \ldots, S_N$ einteilen können in zwei Klassen A und B mit der Eigenschaft, daß keine Seite aus A auf eine Seite aus B verweist und umgekehrt, sind die Gleichungen für die Seiten aus A und die für die Seiten aus B offensichtlich unabhängig voneinander und jedes der beiden Teilsysteme hat nach obiger Diskussion einen mindestens eindimensionales Lösungsraum, und läßt sich jede Lösung des ersten Teilsystems mit jeder Lösung des zweiten zu einer Lösung des Gesamtsystems kombinieren, so daß dessen Lösungsraum mindestens zweidimensional ist. Bei mehr als zwei Klassen, die nur innerhalb der eigenen Klasse zitieren, wird die Dimension noch größer, und für die Praxis am schlimmsten ist die Tatsache, daß uns das Gleichungssystem keinerlei Anhaltspunkte gibt, wie wir die relative Wichtigkeit der einzelnen Klassen festlegen sollen.

Aus diesem Grund muß Gleichung $(*)$ erweitert werden um einen Term, der die Existenz disjunkter Klassen verhindert. Die Idee dazu erklären BRIN und PAGE folgendermaßen:

Gleichung $(*)$ läßt sich auch stochastisch interpretieren: Angenommen, ein Surfer beginnt mit einer zufällig ausgewählten Webseite und klickt, sobald er sie auf dem Bildschirm hat, zufällig auf irgendeinen der dort gefundenen Verweise. Mit der nun erscheinenden Seite verfährt er genauso, und so weiter. Falls dieses Experiment hinreichend oft wiederholt und hinreichend lange durchgeführt wird, ergibt sich als Grenzwert eine Wahrscheinlichkeitsverteilung über alle Webseiten, d.h. wir können für jede Seite S die Wahrscheinlichkeit $p(S)$ ermitteln, daß der Surfer dort ankommt. Offensichtlich genügt auch die Funktion p der Gleichung $(*)$.

Nun wird das Modell etwas modifiziert: Der Surfer klickt nicht mehr unbedingt auf einen der Verweise der aktuellen Webseite, sondern nur mit einer gewissen Wahrscheinlichkeit α. Alternativ geht er mit Wahrscheinlichkeit $1 - \alpha$ zu irgendeiner zufällig ausgewählten Webseite. Jetzt wird die Wahrscheinlichkeit für das Ankommen auf einer Seite S beschrieben durch eine Funktion, die der Rekursionsbedingung

$$p(S) = \alpha \sum_{j=1}^{n} \frac{p(R_j)}{m_j} + \frac{1-\alpha}{N} .$$

Laut BRIN und PAGE ist $\alpha \approx 0{,}85$ eine vernünftige Wahl.

Auch gemäß dieser Rekursionsvorschrift können wir wieder Wichtigkeiten $w(S)$ definieren, die einem linearen Gleichungssystem genügen: Sind $S_1, \ldots, S_N$ die sämtlichen Webseiten (wobei wir annehmen, daß entweder nur Webseiten berücksichtigt werden, die auf andere verweisen, oder aber, daß eine Webseite ohne externe Verweise so behandelt wird, als verwiese sie auf alle Webseiten) und soll S_i die Wichtigkeit w_i bekommen, so muß nun gelten

$$w_i = \alpha \sum_{j=1}^{N} a_{ij} w_j + \frac{1-\alpha}{N} ,$$

wobei die Koeffizienten a_{ij} wie oben definiert sind. Im Gegensatz zum dortigen Ansatz haben wir hier aber für $\alpha \neq 1$ ein inhomogenes lineares Gleichungssystem,

Dieses Gleichungssystem kann für $0 \leq \alpha < 1$ höchstens eine Lösung haben: Sind nämlich $(w_1, \ldots, w_N)$ und $(u_1, \ldots, u_N)$ beides

Lösungsvektoren, so können wir einen Index i finden, für den $|w_i - u_i|$ maximal ist. Für dieses i ist dann

$$
\begin{aligned}
|w_i - u_i| &= \left| \left(\alpha \sum_{j=1}^{N} a_{ij} w_j + \frac{1-\alpha}{N} \right) - \left(\alpha \sum_{j=1}^{N} a_{ij} u_j + \frac{1-\alpha}{N} \right) \right| \\
&= \left| \alpha \sum_{j=1}^{N} a_{ij} (w_j - u_j) \right| \leq \alpha \sum_{j=1}^{N} a_{ij} \left| w_j - u_j \right| \\
&\leq \alpha \sum_{j=1}^{N} a_{ij} \left| w_i - u_i \right| = \left(\alpha \sum_{j=1}^{N} a_{ij} \right) |w_i - u_i| \\
&= \alpha \left| w_i - u_i \right| \ , \ \text{denn} \quad \sum_{j=1}^{N} a_{ij} = 1 \ .
\end{aligned}
$$

Das ist aber nur möglich, wenn $|w_i - u_i|$ verschwindet und damit, wegen dessen Maximaleigenschaft, auch alle anderen Differenzen $w_j - u_j$. Dies zeigt, daß die beiden Lösungen übereinstimmen,

Noch nicht gezeigt ist die *Existenz* einer Lösung, aber jeder, der mit dem BANACHschen Fixpunktsatz vertraut ist, wird wohl wissen, wie es nun weitergeht: Wir starten mit irgendeinem N-tupel $(w_1^{(0)}, \ldots, w_N^{(0)})$ positiver Zahlen und konstruieren dazu sukzessive neue N-tupel gemäß der Vorschrift

$$
w_i^{(k+1)} = \alpha \sum_{j=1}^{N} a_{ij} w_j^{(k)} + \frac{1-\alpha}{N} \ .
$$

Falls das Tupel $(w_1^{(k)}, \ldots, w_N^{(k)})$ eine Lösung ist, stimmt es natürlich mit seinem Nachfolger $(w_1^{(k+1)}, \ldots, w_N^{(k+1)})$ überein, aber das können wir nicht realistischerweise erwarten. Als Maß der Abweichung zwischen den beiden Tupeln betrachten wir das Maximum der Werte $\left| w_i^{(k+1)} - w_i^{(k)} \right|$. Sei $k \geq 1$, und $\left| w_i^{(k)} - w_i^{(k-1)} \right|$ nehme für $i = \ell$ seinen

maximalen Wert an. Dann gilt für alle $i = 1, \dots N$

$$\begin{aligned}
\left| w_i^{(k+1)} - w_i^{(k)} \right| &= \left| \left(\alpha \sum_{j=1}^{N} a_{ij} w_j^{(k)} + \frac{1-\alpha}{N} \right) - \left(\alpha \sum_{j=1}^{N} a_{ij} w_j^{(k-1)} + \frac{1-\alpha}{N} \right) \right| \\
&= \left| \alpha \sum_{j=1}^{N} a_{ij} (w_j^{(k)} - w_j^{(k-1)}) \right| \leq \alpha \sum_{j=1}^{N} a_{ij} \left| w_j^{(k)} - w_j^{(k-1)} \right| \\
&\leq \alpha \sum_{j=1}^{N} a_{ij} \left| w_\ell^{(k)} - w_\ell^{(k-1)} \right| = \left(\alpha \sum_{j=1}^{N} a_{ij} \right) \left| w_\ell^{(k)} - w_\ell^{(k-1)} \right| \\
&= \alpha \left| w_\ell^{(k)} - w_\ell^{(k-1)} \right| , \text{ denn } \quad \sum_{j=1}^{N} a_{ij} = 1 .
\end{aligned}$$

Somit ist

$$\max_i \left| w_i^{(k+1)} - w_i^{(k)} \right| \leq \alpha \max_i \left| w_i^{(k)} - w_i^{(k-1)} \right| ,$$

und da wir $\alpha < 1$ vorausgesetzt haben, werden die Abweichungen immer kleiner. Das CAUCHYsche Konvergenzkriterium zeigt, daß die Folge der $(w_1^{(k)}, \dots, w_N^{(k)})$ konvergiert, und der Limes ist eine Lösung des Gleichungssystems.

Damit ist bewiesen, daß es genau eine Lösung gibt, und nach dem, was wir in der Linearen Algebra gelernt haben, können wir diese mit dem GAUSS-Algorithmus bestimmen.

Es gibt allerdings einen wesentlichen Unterschied zwischen dem hier zu lösenden Gleichungssystem und den aus Übungsblättern und Klausuren bekannten: Zwar geht es in beiden Fällen (meist) um N Gleichungen in N Unbekannten, aber im Studium ist N selten mehr als vier, während es bei Google laut deren Darstellung in *How search works* bei „Hunderten von Milliarden“ liegt.

Um den GAUSS-Algorithmus für ein Gleichungssystem aus N Gleichungen mit N Unbekannten durchzuführen, braucht man asymptotisch etwa N^3 Rechenoperationen. Bei $N \approx 10^{11}$ sind das ungefähr 10^{33}, mit der Näherung $2^{10} \approx 10^3$ also ungefähr 2^{110}.

Bei der Beurteilung der Sicherheit elektronischer Unterschriften geht das Bundesamt für Sicherheit in der Informationstechnik derzeit aus von einem Sicherheitsniveau 2^{100}, das allerdings in den nächsten Jahren auf 2^{120} angehoben werden soll. Ein Verfahren gilt demnach als sicher, wenn anzunehmen ist, daß ein Gegner mindestens 2^{100} *bzw.* 2^{120} Versuche benötigt, um das Verfahren zu knacken. Diese „Versuche" sind zwar etwas komplexer als einfache Rechenoperationen; andererseits muß man aber bei der Beurteilung der Sicherheit von Kryptoverfahren auch Gegner berücksichtigen, die einen Rechenaufwand von einem Jahr oder gar mehr nicht scheuen, was weit jenseits dessen liegt, was für die periodisch zu aktualisierende Rangfolge der Webseiten möglich ist. Daher können wir davon ausgehen, daß 2^{110} Rechenoperationen zumindest für diese Aufgabe derzeit nicht im Bereich des Realisierbaren liegen.

Andererseits sind wir hier auch nicht im Bereich der Reinen Mathematik, sondern es geht um eine Anwendung der Mathematik auf reale Probleme. Dabei müssen wir uns gerade bei so einem Thema auch stets bewußt sein, daß unsere Modelle mit ziemlicher Sicherheit nur eine Approximation der Wirklichkeit sind – falls es hier überhaupt irgendeine „Wirklichkeit" geben sollte.

Von daher wäre es Unsinn, mit riesigem Aufwand ein Problem, das bestenfalls eine grobe Approximation an eine vielleicht gar nicht vorhandene Wirklichkeit beschreibt, mathematisch exakt zu lösen: Eine approximative Lösung reicht vollkommen.

Diese wiederum bietet uns gerade der theoretische Ansatz, mit dem wir die Existenz einer Lösung bewiesen haben: Die dazu verwendete Iteration gestattet uns schließlich eine beliebig genaue Annäherung an die Lösung. Da wir von acht Milliarden Gleichungen in genauso vielen Unbekannten ausgehen, ist schließlich auch die Iterationsvorschrift keine Aufgabe für das Rechnen mit Bleistift und Papier: Um die Gleichung

$$w_i^{(k+1)} = \alpha \sum_{j=1}^{N} a_{ij} w_j(k) + \frac{1-\alpha}{N}$$

für alle i auszuwerten, brauchen wir größenordnungsmäßig N^2 Rechenoperationen *je Iteration.*

Hier hilft uns eine praktische Beobachtung, die wohl keinen Surfer im World Wide Web erstaunen dürfte: Bekanntlich ist $a_{ij} = 0$, wenn die j-te Webseite nicht auf die i-te verweist, und natürlich gibt es kaum Webseiten, die auch nur auf einen Bruchteil aller vorhandener Webseiten verweisen. Experimentelle Untersuchungen zeigen, daß eine Webseite im Durchschnitt nur sieben externe Verweise hat. In der obigen Summe über Hunderte von Milliarden Summanden sind also im Durchschnitt nur ietwa sieben von Null verschieden, wir brauchen also tatsächlich nur etwa $7N$ Additionen und Multiplikationen. Damit sind wir wieder im Bereich der für die langfristige Existenz von Google so wichtigen Skalierbarkeit: Die Zahl N wird natürlich im Laufe der Jahre ziemlich ansteigen, aber zumindest nach bisheriger Erfahrung wird die Rechenkraft pro Dollar (oder Euro) ungefähr im gleichen Maße steigen. Bei der Anzahl sieben für den Durchschnitt für die Verweise auf andere Webseiten sind zumindest mittelfristig keine wesentlichen Änderungen zu erwarten: Die Erzeuger von Webseiten werden wohl auch in Zukunft nicht wesentlich mehr Verweise auf ihren Seiten anbringen, der Aufwand pro Iteration bleibt also ungefähr derselbe, wenn auch künftig der Umfang des World Wide Web im selben Maße ansteigt wie die Rechenkraft der Computer einer festen (inflationsbereinigten) Preisklasse.

Was die Anzahl der Iterationen betrifft, die für ein vorgegebenes Genauigkeitsniveau notwendig sind, sagt uns die Summenformel für geometrische Reihen, daß es nicht auf die Anzahl der Webseiten ankommt: Sei $(w_1, \ldots, w_N)$ die Lösung des linearen Gleichungssystems, $(w_1^{(k)}, \ldots, w_N^{(k)})$ die k-te Iteration und i derjenige Index, für den $\left|w_i - w_i^{(k)}\right|$ maximal ist. Dann ist

$$\begin{aligned}\left|w_i - w_i^{(k)}\right| &= \left|\sum_{\ell=k}^{\infty}(w_i^{(\ell+1)} - w_i^{(\ell)}\right| \leq \sum_{\ell=k}^{\infty}\left|w_i^{(\ell+1)} - w_i^{(\ell)}\right| \\ &\leq \sum_{\ell=0}^{\infty}\alpha^{\ell}\left|w_i^{(k+1)} - w_i^{(k)}\right| = \frac{\left|w_i^{(k+1)} - w_i^{(k)}\right|}{1-\alpha}\end{aligned}$$

nach der Summenformel für die geometrische Reihe. Wir können daher nach jedem Iterationsschritt abschätzen, wie groß der maximale Fehler

ist und abbrechen, sobald dieser eine akzeptable Größenordnung erreicht hat.

In der Arbeit von BRIN und PAGE von 1998

> SERGEY BRIN, LAWRENCE PAGE: *The Anatomy of a Large-Scale Hypertextual Web Search Engine,* Computer networks and ISDN systems, 1998, Elsevier;
> `http://db.stanford.edu/pub/public_html/papers/google.pdf`

ist davon die Rede, daß die Berechnung für das damals untersuchte Netz von 26 Millionen Seiten auf einer (nach damaligen Standards) mittelgroßen *workstation* einige Stunden dauerte. In

> LAWRENCE PAGE, SERGEY BRIN, RAJEEV MOTWANI, TERRY WINOGRAD: *The PageRank Citation Ranking: Bringing Order to the Web,*Technical Report. Stanford InfoLab 1999,
> `http://ilpubs.stanford.edu:8090/422/`

wird das Konvergenzverhalten genauer untersucht und gezeigt, daß für ein Web mit 322 Millionen Links etwa 45 Iterationen ausreichen. Neuere Zahlen werden von Google nicht veröffentlicht; nach externen Schätzungen soll Google einige Tage benötigen, um den PageRank komplett neu zu berechnen.

Die Werte für die Wichtigkeiten können beträchtlich schwanken: der minimale Wert ist offensichtlich gleich $1-d$, also für d = 0,85 gleich 0,15; die theoretische Obergrenze liegt bei dN, also im Milliardenbereich. Google zerteilt diesen Bereich in elf Teilintervalle, denen die PageRanks null bis zehn zugeordnet werden. Über die Definition dieser Intervalle ist nichts bekannt, allerdings wird vermutet, daß die Intervallängen ungefähr in einer geometrischen Progression ansteigen, so daß der PageRank ungefähr gleich einem Logarithmus der gerade bestimmten Wichtigkeit ist, dessen Basis in der Gegend von sechs oder sieben liegen dürfte (6^{10} = 60 466 176 und 7^{10} = 282 475 249). Auch wenn für interne Berechnungen die exakten Werte der w_i verwendet werden, veröffentlicht Google nur die Grobwerte – wahrscheinlich auch dies wieder, um Suchmaschinenoptimierern nicht zuviel Information an die Hand zu geben.

§6: Der HITS-Algorithmus

Zur gleichen Zeit, als BRIN und PAGE in Stanford im Rahmen ihres Dissertationsprojekts den PageRank-Algorithmus entwickelte, war rund zwanzig Kilometer südlich JON KLEINBERG von der Cornell University als *visiting professor* am IBM Almaden Research Center bei San José und befaßte sich ebenfalls mit dem Problem, Webseiten nach Wichtigkeit und Relevanz zu ordnen. Sein Ansatz war etwas komplizierter:

Nach dem Ansatz von BRIN und PAGE gibt eine Webseite mit m Verweisen an jede der aufgeführten Webseiten ein m-tel ihrer eigenen Wichtigkeit weiter, bei großen Werten von m also fast nichts.

Im Falle einer Seite, die wahllos so ziemlich alles zitiert, ist dies sicherlich sinnvoll; gerade damals gab es aber auch noch eine ganze Reihe von Webseiten, auf denen ein oder mehrere Autoren mit großer Mühe eine Sammlung von Referenzen für ein bestimmtes Thema zusammengestellt hatten, teilweise sogar mit Kommentaren zu den einzelnen Seiten. Wenn eine solche Seite gut gemacht ist, verweist sie auf wichtige Seiten, und das sollte bei *deren* Beurteilung auch gebührend gewürdigt werden.

In seiner Arbeit

> JON M. KLEINBERG: *Authoritative sources in a hyperlinked environment,* Journal of the ACM Volume 46 Issue 5, Sept. 1999
> `http://portal.acm.org/citation.cfm?doid=324133.324140`

betrachtet er Webseiten deshalb unter den beiden Gesichtspunkten *hub* und *authority.*

Das englische Wort *hub* hat viele deutsche Übersetzungen; unter anderem bezeichnet es im Luftverkehr ein Drehkreuz, d.h. einen Flughafen, über den eine Gesellschaft beispielsweise die Passagiere ihrer Fernflüge auf die Anschlußflüge zu den weniger zentralen Zielen verteilt, und entsprechend auch die Verteilzentren von Logistikunternehmen. Außerdem steht das Wort für die Nabe eines Rads (von der nach allen Richtungen die Speichen ausgehen), und nicht zuletzt bezeichnet sich auch KLEINBERGs Geburtsstadt Boston als *hub of the universe,* frei übersetzt also als Nabel der Welt.

Auch das Wort *authority* hat viele mögliche Übersetzungen, unter anderem Behörde, Berechtigung, Befehlsgewalt, Ermächtigung, Obrigkeit, Vollmacht; was KLEINBERG meint sind allerdings die alternativen Bedeutungen im Sinne von Fachmann oder Fachkompetenz.

Die Idee ist also, daß *hubs* zentrale Anlaufstellen sind, die auf *authorities* verweisen, die etwas zu einem bestimmten Thema zu sagen haben. Das Prinzip, nach dem er Webseiten ordnet, faßt er in der zitierten Arbeit so zusammen:

> A good *hub* is a page that points to many good authorities; a good *authority* is a page pointed to by many good hubs.

Wie bei den Maximen für den PageRank ist auch diese Definition zirkulär, und wie dort kann die Zirkularität mit Hilfe der Linearen Algebra leicht aufgelöst werden:

KLEINBERG ordnet jeder der Seite S_i *zwei* Gewichte zu: Das *authority*-Gewicht x_i und das *hub*-Gewicht y_i; sie sind so normalisiert, daß der Vektor x mit Komponenten x_i und der Vektor y mit Komponenten y_i jeweils die (EUKLIDische) Länge eins haben. Die obige Maxime wird im wesentlichen so umgesetzt, daß die *authority*-Wichtigkeit einer Seite proportional zur Summe der *hub*-Wichtigkeiten aller darauf verweisender Seiten ist, wohingegen die *hub*-Wichtigkeit proportional zur Summe der *authority*-Wichtigkeiten jener Seiten ist, auf die sie verweist. Die Proportionalitätskonstanten sind jeweils dadurch bestimmt, daß sowohl x als auch y Einheitsvektoren mit nichtnegativen Einträgen sind.

Bezeichnen wir mit A die Matrix mit Einträgen

$$a_{ij} = \begin{cases} 1 & \text{falls es einen Verweis } S_i \to S_j \text{ mit } j \neq i \text{ gibt} \\ 0 & \text{sonst} \end{cases},$$

soll es also Konstanten $\alpha, \beta \in \mathbb{R}_{\geq 0}$ geben, so daß

$$y = \alpha A x \quad \text{und} \quad x = \beta A^T y$$

ist, d.h.

$$x = \alpha\beta A^T A x \quad \text{und} \quad y = \alpha\beta A A^T y\,.$$

Somit ist x ein Eigenvektor von $A^T A$ und y einer von AA^T, beide zum selben Eigenwert.

Tatsächlich geht KLEINBERG nicht von dieser Charakterisierung der beiden Vektoren x und y aus, sondern gibt stattdessen eine Methode zur Konstruktion der beiden Vektoren an: Er startet mit zwei beliebigen Vektoren $x^{(0)}, y^{(0)}$ mit nichtnegativen Einträgen und setzt sukzessive

$$x^{(k)} = A^T y^{(k-1)} \quad \text{und} \quad y^{(k)} = A x^{(k)} .$$

Nach einer gewissen Anzahl von Iterationen, wenn sich die Richtungen von $x^{(k)}$ und $x^{(k-1)}$ sowie die von $y^{(k)}$ und $y^{(k-1)}$ nicht mehr wesentlich voneinander unterscheiden, nimmt er für x den Einheitsvektor in Richtung $x^{(k)}$ und für y den in Richtung $y^{(k)}$.

Dieses Vorgehen erinnert an die Berechnungsmethode für den PageRank. Um zu sehen, ob und wohin KLEINBERGs Folgen konvergieren, beachten wir, daß

$$x^{(k)} = A^T A x^{(k-1)} \quad \text{und} \quad y^{(k)} = A A^T y^{(k-1)}$$

ist; wir müssen uns also nur überlegen, wohin für eine symmetrische Matrix M die durch $z^{(k)} = M z^{(k-1)}$ definierte Folge für einen gegebenen Anfangsvektor $z^{(0)}$ konvergiert.

Wie wir wissen, gibt es zu einer symmetrischen Matrix eine Basis aus Einheitsvektoren, bezüglich derer sie Diagonalgestalt hat; durch Umordnen erhalten wir eine Basis $(b^{(1)}, \ldots, b^{(N)})$, bezüglich derer M Diagonalgestalt hat mit Diagonaleinträgen $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N$. Für $z^{(0)} = z_1 b^{(1)} + \cdots + z_N b^{(N)}$ mit $z_i > 0$ ist dann

$$z^{(k)} = \lambda_1^k z_1 b^{(1)} + \cdots + \lambda_N^k z_n b^{(N)} .$$

Für alle $\lambda_i < \lambda_1$ geht der Quotient $\lambda_i^k / \lambda_1^k$ gegen null; im Einheitsvektor zu $z^{(k)}$ kommen daher für große Werte von k praktisch nur noch die Koeffizienten vor, die mit λ_1^k multipliziert wurden. Ist $\lambda_1 > \lambda_2$, bekommen wir somit als Ergebnis nach Normalisierung den Eigenvektor $b^{(1)}$ zum größten Eigenwert λ_1; ansonsten erhalten wir einen von $z^{(0)}$ abhängigen Vektor aus dem Eigenraum zum Eigenwert λ_1. Wie im Falle von PageRank kann man letzteres ausschließen, indem man die Matrix M ersetzt durch eine konvexe Linearkombination mit der Matrix, deren sämtliche

Einträge eins sind, allerdings zeigen experimentelle Untersuchungen, daß man auch ohne diese Maßnahme auskommt.

Das Verfahren von KLEINBERG unterscheidet sich noch in einem zweiten Punkt von PageRank: Dort werden bekanntlich sämtliche von den Crawlern gefundene Webseiten unabhängig von jeder Suchanfrage global nach ihrer Wichtigkeit geordnet. KLEINBERG dagegen betrachtet nur einen Ausschnitt des Webs: In der oben zitierten Originalarbeit schlägt er vor, beispielsweise mit den ersten zweihundert Ergebnissen der damals populären Suchmaschine Altavista zu starten; in heutigen Darstellungen ist die Rede davon, mit allen bekannten Seiten anzufangen, die die Suchbegriffe enthalten.

In einem nächsten Schritt wird diese Ausgangsmenge erweitert um alle Webseiten, die entweder einen Verweis auf eine der betrachteten Seiten enthalten oder aber Ziel eines Links von einer derartigen Seite sind. (Dieses Verfahren kann man gegebenenfalls noch ein oder mehrere Male wiederholen.) Danach wird die Matrix A für das so erhaltene Teilnetz aufgestellt, und nur für dessen Seiten werden die *hub*- und *authority*-Wichtigkeiten aufgestellt. Man beachte, daß durch die Erweiterung der ursprünglichen Menge von Webseiten auch Seiten einbezogen werden und möglicherweise sogar hohe Wichtigkeiten bekommen, die überhaupt keinen der Suchbegriffe enthalten. Auf diese Weise erreicht der Algorithmus auch ohne Untersuchung der Term-Dokument-Matrix eine Art latente semantische Analyse.

§7: Die Gewichte der Terme

Kehren wir zurück zur Term-Dokument-Matrix. Ihr Eintrag a_{ij} soll eine Art Gewicht des i-ten Term im j-ten Dokument sein. Im einfachsten Fall nimmt a_{ij} nur die beiden Werte 0 und 1 an, je nachdem ob der der Begriff im Dokument vorkommt oder nicht; eine andere offensichtliche Lösung wäre, daß a_{ij} zählt, wie oft der Begriff auftritt. Beides ist zwar einfach, führt aber auch zu offensichtlichen Problemen: Nicht jedes Wort, das irgendwo in einem Dokument vorkommt, läßt auf den Inhalt des Dokuments schließen, und wenn ein Wort sehr häufig vorkommt,

kann das auch einfach nur bedeuten, daß das Dokument entweder sehr lang oder sehr geschwätzig ist.

Seit es Textdatenbanken gibt werden daher auch kompliziertere Schemata diskutiert und immer weiter verfeinert; Google etwa benutzt nach eigenen Angaben rund zweihundert sogenannte „Signale“, um die Relevanz eines Dokuments für eine Suchanfrage zu bestimmen.

Schon bei deutlich einfacheren Vorgehensweisen empfiehlt es sich, zwischen der *lokalen* und der *globalen* Wichtigkeit eines Begriffs zu unterscheiden. Dabei soll die lokale Wichtigkeit messen, welche relative Bedeutung ein Begriff innerhalb eines speziellen Dokuments hat, während die globale Wichtigkeit angibt, wie wichtig der Begriff für die gesamte Dokumentensammlung ist. Der Eintrag in der Term-Dokument-Matrix ist das Produkt der beiden Wichtigkeiten, wobei anschließend eventuell noch alle Spalten in geeigneter Weise normalisiert werden.

Beginnen wir mit der lokalen Wichtigkeit. Die beiden einfachsten Schemata wurden bereits erwähnt: Wir setzten die lokale Wichtigkeit ℓ_{ij} des i-ten Begriffs für das j-te Dokument entweder nur auf 0 oder 1, je nachdem ob der Begriff im Dokument vorkommt, oder aber wir setzten ℓ_{ij} auf die Anzahl f_{ij} der Vorkommen des Begriffs im Dokument. Um die damit verbundene Bevorzugung langer Dokumente abzumildern, wird teilweise auch der Logarithmus verwendet; da dieser für den Wert null nicht definiert ist, setzt man hier

$$\ell_{ij} = \log(1 + f_{ij}) \,.$$

Um völlig unabhängig von der Dokumentlänge zu werden, kann man auch die *durchschnittliche* Häufigkeit f_j der Terme im j-ten Dokument betrachten: Ist u_j die Anzahl verschiedener Terme im Dokument, so setzten wir

$$f_j = \frac{1}{u_j} \sum_{i=1}^{m} f_{ij} \quad \text{und} \quad \ell_{ij} = \frac{\log(1 + f_{ij})}{\log(1 + f_j)} \,.$$

Hier ist $\ell_{ij} = 1$ für jeden Begriff, der genau die mittlere Häufigkeit hat: kommt der Term überdurchschnittlich oft vor, ist $\ell_{ij} > 1$, ansonsten kleiner.

Ein Kompromiss zwischen bloßem Vorkommen und (relativer) Häufigkeit ist die bereits von SALTON vorgeschlagene vergrößerte normalisierte Häufigkeit

$$\ell_{ij} = \frac{1}{2}\left(\chi(f_{ij}) + \frac{f_{ij}}{\max_\nu f_{\nu j}}\right) \quad \text{mit} \quad \chi(x) = \begin{cases} 1 & \text{falls } x \neq 0 \\ 0 & \text{falls } x = 0 \end{cases}.$$

Die globale Wichtigkeit g_i hängt nur vom Suchbegriff ab. Durch sie soll berücksichtigt werden, daß eher seltene Begriffe meist deutlich spezifischer sind als Allerweltsbegriffe, die in praktisch jedem Dokument vorkommen. Das extremste Beispiel dafür sind die bereits erwähnten *Nullen* auf der Stopliste, die überhaupt nicht berücksichtigt werden; im Rahmen der jetzigen Betrachtungsweise können wir sie definieren als die Wörter mit $g_i = 0$.

Ein Begriff ist unter dem Gesichtspunkt der Informationssuche umso spezifischer, je ungleichmäßiger er über die Dokumente verteilt ist. Da die SHANNONsche Entropie derartige Ungleichmäßigkeiten quantifiziert, liegt es nahe, sie auch für die Definition einer globalen Wichtigkeit einzusetzen. Wir betrachten alle Vorkommen des i-ten Begriffs in den n Dokumenten der Sammlung und definieren als

$$p_{ij} = \frac{f_{ij}}{\sum_{j=1}^{n} f_{ij}}$$

den Anteil dieser Vorkommen im j-ten Dokument. Die Summe

$$-\sum_{j=1}^{n} p_{ij} \log p_{ij}$$

hat ihren maximalen Wert $\log n$, wenn der Begriff in jedem Dokument gleich häufig auftritt; den minimalen Wert Null nimmt sie an, wenn er nur in einem einzigen Dokument vorkommt. Damit bietet sich

$$g_i = 1 + \frac{\sum_{j=1}^{n} p_{ij} \log p_{ij}}{\log n}$$

als eine Möglichkeit zur Definition der globalen Wichtigkeit an: Im Falle der gleichmäßigen Verteilung erhalten wir den Wert Null, für einen Begriff, der nur in einem einzigen Dokument vorkommt dagegen den maximal möglichen Wert eins.

Für ein einfacheres Maß können wir auch einfach nur zählen, in wie vielen Dokumenten der Begriff vorkommt. Ist n_i diese Anzahl, so ist

$$g_i = \log \frac{n}{n_i}$$

gleich Null für einen Begriff, der in jedem Dokument vorkommt, wohingegen der Maximalwert $\log n$ angenommen wird. falls das Wort nur in einem Dokument steht. Diese sogenannte *inverse Dokumenthäufigkeit* wird vor allem gerne eingesetzt für Sammlungen, deren Inhalt sich nicht allzu häufig ändert. Alternativ wird auch gelegentlich das sogenannte probabilistische Inverse

$$g_i = \log \frac{n - n_i}{n_i}$$

verwendet, das die Anzahlen von Dokumenten mit *bzw.* ohne den Begriff zueinander in Beziehung setzt.

Eine völlig andere Strategie besteht darin, die Wichtigkeit eines Begriffs danach zu beurteilen, wie oft er in den Dokumenten auftritt, in denen er überhaupt vorkommt; damit hätten wir also

$$g_i = \frac{1}{n} \sum_{j=1}^{n} f_{ij} \, .$$

Dieses Maß ist offensichtlich nur sinnvoll, wenn Nullen vorher ausgeschlossen wurden, denn es würde auch beispielsweise für bestimmte Artikel eine hohe Wichtigkeit liefern.

Möchte man die globalen Wichtigkeiten nach Seltenheit des Suchbegriffs festlegen, bietet sich auch an, den Vektor $(f_{i1}, \ldots, f_{in}) \in \mathbb{R}^n$ der Anzahlen zu betrachten; da seltene Begriffe kurzen Vektoren entsprechen, kann die globale Wichtigkeit als Kehrwert

$$g_i = \frac{1}{\sqrt{\sum_{j=1}^{n} f_{ij}^2}}$$

der EUKLIDischen Länge definiert werden.

Durch Multiplikation der lokalen und globalen Wichtigkeiten erhält man erhält man ein Maß für die Relevanz des i-ten Terms im j-ten

Dokument. Bei den meisten Wahlen erhält man dabei Werte, die lange Dokumente gleich in zweierlei Hinsicht bevorzugen: Einmal stehen in einem langen Dokument im allgemeinen mehr verschiedene Begriffe; die Wahrscheinlichkeit, daß ein Begriff aus der Suchanfrage überhaupt vorkommt, ist also größer. Zum andern wird ein fester Begriff, so er überhaupt vorkommt, in einem langen Dokument meist häufiger vorkommen als in einem kurzen.

Diesen Effekt kann man durch Normalisierung abmildern oder sogar aufheben. Wir kennen bereits die häufig angewendete Strategie, den Kosinus des Winkels zwischen dem Spaltenvektor des Dokuments und dem Anfragevektor als Ähnlichkeitsmaß zu verwenden; dies entspricht der Normierung der Spalten auf EUKLIDische Länge eins und der Skalarproduktbildung zur Berechnung der Relevanz. Wie experimentelle Untersuchungen zeigen, führt diese Strategie zu einer Bevorzugung *kurzer* Dokumente. Auch der Vergleich mit Mittel- oder Maximalwerten kann zur Normierung eingesetzt werden – ein Beispiel haben wir bereits oben bei den lokalen Gewichten betrachtet.

In der Arbeit

> AMIT SINGHAL. CHRIS BUCKLEY, MANDAR MITRA: *Pivoted Document Length Normalization,* Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval, 1976
> `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.50.9950&rep=rep1&type=pdf`

wird untersucht, wie für fünfzig Suchanfragen an eine Datenbank mit 741 856 Dokumenten einerseits, wie jeweils einerseits die Anzahl der relevanten Dokumente und andererseits die der gefundenen Dokumente von der Dokumentlänge abhängt. Als Ergebnis erhielten die Autoren zwei Kurven, die sich in einem bestimmten Punkt schneiden; vor diesem Punkt liegt die eine Kurve oben, danach die andere.

Idealerweise sollten natürlich beide Kurven übereinstimmen; die Übereinstimmung kann verbessert werden, indem durch geeignete Renormierung die Kurve der gefundenen Dokumente um den Schnittpunkt

so gedreht wird, daß die Tangenten beider Kurven dort übereinstimmen. Dazu werden verschiedene Ansätze diskutiert, beispielsweise die Definition

$$a_{ij} = \frac{(1 + \log f_{ij})/(1 + log\overline{f}_j)}{(1 - s)p + su_j} ,$$

wobei wieder $\overline{f}_j$ die mittlere Anzahl der Vorkommen eines Worts im j-ten Dokument ist, p ist die mittlere Anzahl verschiedener Begriffe pro Dokument und u_j die Anzahl verschiedener Wörter in Dokument j. Der *Slope* s definiert den Winkel, um den gedreht wird, z.B. $s \approx 0{,}2$. Damit erhielten sie um 13,7% bessere Ergebnisse als mit der üblichen Kosinusstrategie.

Natürlich sind die hier vorgestellten Maße nur ein kleiner Ausschnitt aus der Vielfalt aller denkbarer Möglichkeiten, und es sind vor allem die in der akademischen Welt diskutierten. Die kommerziell erfolgreichen Suchmaschinen und sonstigen Textverwaltungssysteme dürften wohl deutlich kompliziertere Maße verwenden, deren Einzelheiten sie aus gutem Grund für sich behalten. Publizierte experimentelle Tests gibt es im wesentlichen nur für kleine Systeme; ihre Ergebnisse lassen sich nur sehr bedingt auf große Zeitschriftendatenbanken oder gar das gesamte World Wide Web übertragen – insbesondere da im letzteren mittlerweile viele Webseitenoptimierer damit beschäftigt sind, Rangfolgen zu analysieren und die Seiten ihrer Kunden nach vorne zu bringen. Bevor das Problem der optimalen Gewichtung wirklich verstanden ist, sind sicherlich noch viele theoretische wie auch experimentelle Studien notwendig.

§8: Mehr über Matrixzerlegungen

Wir haben bislang nur die Singulärwertzerlegungeiner Matrix betrachtet und gesehen, daß wir mit ihrer Hilfe die im Sinne der FROBENIUS-Norm nächstgelegene Matrix finden können, deren Rang eine vorgegebene Schranke nicht überschreitet; dies war der Ausgangspunkt zur latenten semantischen Analyse.

Sowohl in der theoretischen Literatur als auch in praktisch implementierten Systemen werden daneben noch eine ganze Reihe weiterer Zer-

legungen diskutiert *bzw.* angewendet, die teils einfacher zu berechnen sind (die Singulärwertzerlegungder Term-Dokument-Matrix des gesamten *World Wide Web* ist mit den heute zur Verfügung stehenden Computern und Algorithmen definitiv nicht berechenbar), teils auch theoretische oder – zumindest für gewisse Anwendungen – auch experimentell gezeigte praktische Vorteile haben.

Einige dieser Zerlegungen sollen hier zumindest kurz diskutiert werden; für eine ausführliche Betrachtung sei auf das Buch

> DAVID SKILLICORN: Understanding Complex Datasets – Data Mining with Matrix Decompositions, *Chapman & Hall/CRC,* 2007

verwiesen.

a) Allgemeines über Matrixzerlegungen

Genau wie bei der Singulärwertzerlegungbetrachten wir Produktdarstellungen (oder Approximationen davon) einer Matrix A in der Form

$$A = UDV \quad \text{oder} \quad A \approx UDV\,.$$

Dabei soll U eine $m \times r$-Matrix sein, V eine $r \times n$-Matrix und D eine $r \times r$-Diagonalmatrix. Bei der zweiten Form soll die Matrix UDV in irgendeinem Sinne einfacher sein als A, aber sich nicht allzu wesentlich von A unterscheiden. Im Falle der Singulärwertzerlegungetwa könnten wir hier die im Sinne der FROBENIUS-Norm bestmögliche Approximation durch eine Matrix nehmen, deren Rang eine vorgegebene Schranke nicht übersteigt. Da r in den meisten Fällen deutlich kleiner ist als n und m, nehmen die drei Faktoren U, D, V zusammen oft weniger Speicherplatz in Anspruch als die Ausgangsmatrix A. Dies gilt insbesondere dann, wenn A nur *ungefähr* gleich der rechten Seite ist, da wir dann (wie bereits bei der latenten semantischen Analyse via Singulärwertzerlegung) r noch einmal reduzieren.

Interpretieren wir A als die Term-Dokument-Matrix einer Textsammlung oder einer Suchmaschine, so entsprechen die n Spalten von A den vorhandenen Dokumenten, die m Zeilen den möglichen Suchbegriffen.

Auch V hat n Spalten; daher sollten auch diese etwas mit den Dokumenten zu tun haben. Während die Spalten von A Vektoren im $\mathbb{R}^m$ sind, liegen die Spalten von V in $\mathbb{R}^r$, liefern also eine andere Beschreibung. Die Komponenten der Spaltenvektoren von A entsprechen den Wichtigkeiten der Suchterme innerhalb des Dokuments; die Komponenten des entsprechenden Spaltenvektors von V stehen in einem linearen Zusammenhang damit. Bei einer guten Matrixzerlegung sollten sie der Wichtigkeit von „Konzepten“ innerhalb des Dokuments entsprechen; man spricht gelegentlich auch von „latenten Begriffen“.

Die Matrix U hat dieselbe Anzahl von Zeilen wie A, und die Zeilen von A entsprechen den möglichen Suchbegriffen. Es liegt daher nahe, die Spalten von U als „latente Dokumente“ zu betrachten.

Nach den Regeln der Matrixmultiplikation ist

$$a_{ij} = \sum_{k=1}^{r} u_{ik} d_{kk} v_{kj} \; ;$$

der Eintrag a_{ij} der Term-Dokument-Matrix ist also eine Art gewichtetes Skalarprodukt aus dem i-ten Zeilenvektor von U und dem j-ten Spaltenvektor von V; dabei wird der k-te Summand mit d_{kk} gewichtet. Im j-ten Spaltenvektor von V stehen die Wichtigkeiten der latenten Begriffe für das j-te Dokument; jede von diesen wird mit einem globalen, nur vom latenten Begriff abhängigen Gewichtsfaktor aus der Diagonalmatrix D multipliziert. Die Einträge des i-Zeilenvektors von U schließlich können wir dann interpretieren als Gewichtsfaktoren, die die Wichtigkeiten der latenten Begriffe kombinieren zur Wichtigkeit des i-ten Terms.

Diese Interpretationen sind natürlich nur das, was wir gerne hätten; *a priori* spricht nichts dafür, daß eine vorgegebene Matrixzerlegung sinnvoll auf diese Weise interpretiert werden kann, daß also die „latenten Begriffe“ zumindest ungefähr sinnvollen Konzepten entsprechen, an denen Anwender interessiert sind. Bei der Singulärwertzerlegunghatten wir eine gewisse Heuristik (die Rangreduktion), die so etwas wahrscheinlich machte, aber hier wie bei anderen Zerlegungen kann letztendlich nur der praktische Einsatz zeigen, wie erfolgreich sie wirklich sind.

Im folgenden seien einige Zerlegungen, die zumindest versuchsweise schon eingesetzt wurden, kurz vorgestellt.

b) Die QR-Zerlegung

Diese Zerlegung wird in der Linearen Algebra häufig verwendet und ist beispielsweise auch die Grundlage einiger numerischer Verfahren zur Bestimmung der Eigenwerte und Eigenvektoren einer Matrix. Sie hat die Form $A = QR$ oder $A = Q_1 R_1$, wobei in der ersten Fom Q eine orthogonale $n \times n$-Matrix ist und R eine $n \times m$-Matrix, deren Einträge r_{ij} für $i > j$ verschwinden; im Falle $n = m$ ist R also eine obere Dreiecksmatrix. Die Diagonalmatrix D ist in diesem Fall die Einheitsmatrix.

Die Spalten von Q bilden eine Orthonormalbasis des $\mathbb{R}^n$; falls der von den ersten ℓ Spalten von A aufgespannte Untervektorraum des $\mathbb{R}^n$ die Dimension k hat, sollen die ersten k Spalten von Q eine Basis dieses Untervektorraums sein.

Für den Fall, daß der Rang r von A kleiner als n ist, werden die Spalten von Q hinter der r-ten sowie die Zeilen von R unter der r-ten nicht gebraucht, um die Spalten von A zu erzeugen; daher reicht es, die $n \times r$-Matrix Q_1 aus den ersten r Spalten von Q zu betrachten und die $r \times m$-Matrix R_1 aus den ersten r Zeilen von R; dies ist die zweite, kompaktere Form der Zerlegung.

In beiden Formen kann die QR-Zerlegung relativ einfach berechnet werden durch jedes Verfahren, das zu einer gegebenen Basis eines Untervektorraums eine Orthonormalbasis liefert, also beispielsweise nach GRAM-SCHMIDT. Da dieses Verfahren allerdings numerisch eher instabil ist, verwendet man beim numerischen Rechnen meist alternative Verfahren wie HOUSEHOLDER-Transformationen oder GIVENS-Rotationen.

Da Q eine orthogonale Matrix ist, ändert die Multiplikation mit Q nichts an den Winkeln; insbesondere ist der Winkel zwischen dem k-ten Spaltenvektor von A und v gleich dem Winkel zwischen dem k-ten Spaltenvektor von R_1 und $Q^{-1}v = Q^T v$. Zusätzlich läßt sich mit der QR-Zerlegung auch eine Rangreduktion erreichen, indem man Zeilen von R, in denen nur betragskleine Zahlen stehen, durch Nullzeilen ersetzt.

c) Die semidiskrete Zerlegung

Diese Zerlegung wird meist als näherungsweise Zerlegung

$$A \approx UDV^T$$

berechnet, wobei U für eine $n \times m$-Matrix A eine $n \times r$-, V eine $m \times r$- und D eine $r \times r$-Matrix ist. Dabei soll D wie üblich eine reelle Diagonalmatrix sein; von den Matrizen U und V verlangen wir, daß sie nur 0 und ± 1 als Einträge haben dürfen. Wegen dieser gravierenden Einschränkung muß hier r für eine gute Approximation oder gar Gleichheit oft größer sein als n und m; aus dem gleichen Grund können U und V auch mehrere identische Spalten oder Zeilen enthalten.

Es ist natürlich stets möglich, eine semidirekte Zerlegung mit Gleichheitszeichen zu finden; spätestens mit $r = nm$ geht das für jede Matrix: Ist $u_1, \ldots, u_n$ eine Basis von $\mathbb{R}^n$ und $v_1, \ldots, v_m$ eine von $\mathbb{R}^m$, so bilden die Vektoren $u_i \otimes v_j$ eine Basis des Raums aller $n \times m$-Matrizen, d.h es gibt eine Darstellung

$$A = \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij} u_i \otimes v_j \; .$$

Bezeichnet D die $nm \times nm$- Diagonalmatrix, in der die Zahlen d_{ij} in der Reihenfolge

$$d_{11}, d_{12}, \ldots, d_{1m}, d_{21}, \ldots d_{2m}, \ldots, d_{n1}, \ldots, d_{nm}$$

in der Diagonale stehen, so ist $A = UDV^T$ für die $n \times nm$-Matrix U, in der die ersten m Spalten gleich dem Vektor u_1 sind, die nächsten m gleich u_2, und so weiter, und V ist die Matrix, deren Spalten aus n aufeinanderfolgenden Gruppen $v_1, \ldots, v_m$ bestehen. Für eine allgemeine 2×3-Matrix ist beispielsweise $\begin{pmatrix} a & b & c \\ d & e & f \end{pmatrix}$ gleich

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} a & 0 & 0 & 0 & 0 & 0 \\ 0 & b & 0 & 0 & 0 & 0 \\ 0 & 0 & c & 0 & 0 & 0 \\ 0 & 0 & 0 & d & 0 & 0 \\ 0 & 0 & 0 & 0 & e & 0 \\ 0 & 0 & 0 & 0 & 0 & f \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} ,$$

Diese Darstellung verrät uns natürlich nichts Neues über die Matrix A. Interessant ist die semidiskrete Zerlegung einer Matrix daher nur, wenn wir mit deutlich weniger als nm Faktoren auskommen.

Die Idee, die hinter einer solchen Zerlegung steckt, wird am klarsten bei einer geometrischen Interpretation der Matrix A: Wir betrachten für eine $n \times m$-Matrix A eine Rechteck mit Seitenlängen n und m, das wir in nm Quadrate unterteilen. Auf jedem dieser Quadrate errichten wir einen Turm; der auf dem Quadrat (i, j) hat die Höhe $\left|a_{ij}\right|$ und geht je nach dem Vorzeichen von a_{ij} entweder nach oben oder nach unten. Diese Türme wollen wir nun so schnell wie möglich abbauen.

Im ersten Schritt ist $r = 1$. Wir suchen zwei Vektoren $u^{(1)} \in \{-1, 0, 1\}^n$ und $v^{(1)} \in \{-1, 0, 1\}^m$, die wir mit einspaltigen Matrizen identifizieren, und eine reelle Zahl d_1 derart, daß Subtraktion der Matrix

$$d_1 u^{(1)} \otimes v^{(1)} = d_1 u^{(1)} v^{(1)T}$$

eine vorher festzulegende Norm maximal reduziert. Wenn es um eine maximale Volumenreduktion der Türme geht, bietet sich hier die Summe der Beträge aller Einträge der Matrix an, je nach Anwendung können aber die FROBENIUS-Norm oder die L^2-Norm geeigneter sein.

Die Matrizen $u \otimes v = uv^T$ sind von einer sehr speziellen Form: Der (i, j)-Eintrag ist genau dann von Null verschieden, wenn weder u_i noch v_j verschwinden; da beide dann ± 1 sein müssen, ist der Eintrag dann eins, falls u_i und v_j das gleiche Vorzeichen haben, und -1 sonst. Falls die i mit $u_i \neq 0$ und die j mit $v_j \neq 0$ fortlaufende Teilfolgen der natürlichen Zahlen bilden, liegen die $(i, j$, für die der Eintrag von $u \otimes v$ nicht verschwindet, in einem Teilrechteck; im allgemeinen Fall haben wir eine unzusammenhängende Menge von Teilrechtecken.

Betrachten wir als einfaches Beispiel der Matrix

$$A = \begin{pmatrix} 4 & 1 & 2 \\ 2 & 3 & 4 \end{pmatrix} .$$

Die höchsten „Türme“ sind die beiden Vierer in den Ecken. Die sie weder in einer gemeinsamen Zeile noch einer gemeinsamen Spalte stehen, gibt es allerdings keine Vektoren u, v, so daß $u \otimes v$ genau an diesen beiden

Stellen von Null verschieden ist. Das geht erst, wenn wir alle vier Ecken nehmen:

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

Als Kompromiss zwischen den Vierern und Zweiern in den vier Ecken können wir die Höhe drei wählen und damit $u^{(1)} = \binom{1}{1}$, $v^{(1)} = (\,1 \quad 0 \quad 1\,)^T$ und $d_1 = 3$ Wählen. Dann ist

$$A - 3u^{(1)} \otimes v^{(1)} = \begin{pmatrix}1 & 1 & -1\\-1 & 3 & 1\end{pmatrix} .$$

In einem zweiten Schritt können wir die Ecken vollständig eliminieren, denn mit $u^{(2)} = \binom{1}{1}$ und $v^{(2)} = (\,1 \quad 0 \quad -1\,)^T$ ist

$$u^{(2)} \otimes v^{(2)} = \begin{pmatrix}1 & 0 & -1\\-1 & 0 & 1\end{pmatrix}$$

und

$$A - \begin{pmatrix}1 & 1\\1 & 1\end{pmatrix}\begin{pmatrix}3 & 0\\0 & 1\end{pmatrix}\begin{pmatrix}1 & 1\\0 & 0\\1 & -1\end{pmatrix}^T = \begin{pmatrix}0 & 1 & 0\\0 & 3 & 0\end{pmatrix} .$$

Wenn diese Abweichung noch zu groß ist, können wir in einem nächsten Schritt die Drei eliminieren durch Subtraktion von $3\binom{0}{1} \otimes (\,0 \quad 1 \quad 0\,)^T$ und erhalten

$$A = \begin{pmatrix}1 & 1 & 0\\1 & 1 & 1\end{pmatrix}\begin{pmatrix}3 & 0 & 0\\0 & 1 & 0\\0 & 0 & 3\end{pmatrix}\begin{pmatrix}1 & 1 & 0\\0 & 0 & 1\\1 & -1 & 0\end{pmatrix} + \begin{pmatrix}0 & 1 & 0\\0 & 0 & 0\end{pmatrix} .$$

Den verbleibenden Fehler können wir, falls gewünscht, in einem vierten Schritt eliminieren, um so eine exakte Zerlegung

$$A = \begin{pmatrix}1 & 1 & 0 & 1\\1 & 1 & 1 & 0\end{pmatrix}\begin{pmatrix}3 & 0 & 0 & 0\\0 & 1 & 0 & 0\\0 & 0 & 3 & 0\\0 & 0 & 0 & 1\end{pmatrix}\begin{pmatrix}1 & 1 & 0 & 1\\0 & 0 & 1 & 0\\1 & -1 & 0 & 0\end{pmatrix}$$

zu erhalten. Dies ist natürlich nicht die einzig mögliche Zerlegung; neben der oben aufgeführten Zerlegung einer allgemeinen 2×3-Matrix mit $r = 6$ gibt es auch noch eine weitere Zerlegung mit $r = 4$: Wenn wir im

dritten Schritt von der Spalte $\binom{1}{3}$ die Spalte $\binom{2}{2}$ subtrahieren, bleibt als Rest in der mittleren Spalte $\binom{-1}{1}$ übrig, die wir auch mit einem einzigen Schritt eliminieren können:

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & -1 \end{pmatrix} \begin{pmatrix} 3 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & -1 & 0 & 0 \end{pmatrix}$$

Die semidiskrete Zerlegung unterscheidet sich in einem wesentlichen Punkt von der Singulärwertzerlegung: Jede Summand der Singulärwertzerlegungeiner Term-Dokument-Matrix $A = \sum \sigma_i u^{(i)} \otimes v^{(i)}$ entspricht einer Kollektion von Termen und beschreibt deren Vorkommen in allen Dokumenten. Bei der semidiskreten Zerlegung $A = \sum d_i u^{(i)} \otimes v^{(i)}$ dagegen geht es um Summanden, die sich nur jeweils auf einen Teil der Dokumente und einen Teil der darin vorkommenden Terme beziehen. Die Zerlegung ist somit in einem gewissen Sinne feiner. Trotzdem ist ihr Speicherbedarf verhältnismäßig gering, da Matrizen, deren Einträge nur $0, 1$ und -1 sind, sehr kompakt abgespeichert werden können. Tatsächlich stammt die Zerlegung ursprünglich aus der Bildverarbeitung und wurde vor allem deswegen eingeführt.

Bei der Informationssuche kann die semidiskrete Zerlegung zur latenten semantischen Analyse benutzt werden, da die Summanden etwas aussagen über die Zugehörigkeit potentieller Suchbegriffe zu Themengebieten. Sie kann aber auch in Kombination mit der Singulärwertzerlegungangewandt werden, indem man zunächst mit der SVD „entrauscht“ und dann mit der SDD die Inhalte Themengebieten zuordnet.

Algorithmisch geht man zur approximativen semidiskreten Zerlegung einer $n \times m$-Matrix A mit Parameter r folgendermaßen vor: Zunächst wird eine Matrixnorm $\|\cdot\|$ gewählt, und die „Restmatrix“ R wird auf A initialisiert. In den folgenden r Schritten sucht man jeweils nach Vektoren $u^{(i)} \in \{-1, 0, 1\}^n$ und $v^{(i)} \in \{-1, 0, 1\}^m$ sowie einer positiven reellen Zahl d_i derart, daß $\left\| R - d_i u^{(i)} \otimes v^{(i)} \right\|$ minimal wird. Für ein solches Tripel wählt man $u^{(i)}$ und $v^{(i)}$ als i-te Spalte von U *bzw.* V und d_i als i-ten Diagonaleintrag von D; außerdem wird R ersetzt durch $R - d_i u^{(i)} \otimes v^{(i)}$.

Die Minimierungsprobleme in den r Schritten sind recht aufwendig; üblicherweise begnügt man sich daher mit der folgenden Annäherung: Man startet mit irgendeinem Vektor $v \in \{-1, 0, 1\}^m$ und sucht dann (mit klassischen Methoden der ganzzahligen Optimierung) nach einem Vektor u und einer Zahl d, die die Norm minimieren. Sodann sucht man für den gefundenen Vektor u nach v und d, die die Norm minimieren, und so weiter, bis das Ergebnis hinreichend stabil ist.

d) Die nichtnegative Zerlegung

Auch hier ist wieder die Diagonalmatrix D gleich der Einheitsmatrix; wir suchen also nach Zerlegungen der Form $A = WH$, wobei W und H nur nichtnegative Einträge haben dürfen. Bezüglich der allgemeinen Interpretation einer Matrixzerlegung aus Abschnitt *a)* bedeutet dies, daß wir unsere latenten Begriffe und latenten Dokumente ohne negative Koeffizienten aufbauen. Es gibt verschiedene Ansätze zur Berechnung solcher Zerlegungen; einige davon haben (im Gegensatz zu den anderen bislang betrachteten Zerlegungen) die schöne Eigenschaft, daß für eine spärlich besetzte Matrix (was die Term-Dokument-Matrix praktisch immer ist) auch die Faktoren spärlich besetzt sind.

§9: Ausblick: Alternative Methoden

Natürlich gibt es außer den in diesem Kapitel diskutierten Methoden noch eine ganze Reihe weiterer mathematischer Ansätze zur Klassifikation von Dokumenten. In diesem letzten Paragraphen sollen beispielhaft drei skizziert werden:

a) Clustering

Hier geht es um ein rein kombinatorisch-statistisches Verfahren. Eine endliche Datenmenge D sei gegeben als Teilmenge eines $\mathbb{R}^n$, und auf $\mathbb{R}^n$ sei irgendeine Abstandsfunktion $d\colon \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ definiert. Das kann natürlich der übliche EUKLIDische Abstand sein, es könnte aber im Falle einer Menge D von Spalten einer Term-Dokument-Matrix auch der (ungerichtete) Winkel zwischen zwei Vektoren sein. Ziel ist es, die Menge D

als disjunkte Vereinigung von Teilmengen C_i, den sogenannten Clustern, darzustellen, wobei die Elemente jeder dieses Clusters jeweils nahe beieinander liegen sollen.

Ist $C = \{x_1, \ldots, x_m\} \subseteq D \subset \mathbb{R}^n$ ein Cluster, so definieren wir den Schwerpunkt von C als

$$S_C \underset{\text{def}}{=} \frac{1}{m} \sum_{j=1}^{m} x_j$$

und den Abstand zweier Cluster als den durch die Funktion d ausgedrückten Abstand ihrer Schwerpunkte.

Ein einfacher Algorithmus zur Clusterung besteht darin, daß man zunächst jeden Punkt aus zu einem einelementiger Cluster macht. Danach sucht man sukkzessive in jedem Schritt zwei Cluster mit minimalem Abstand und vereinigt diese zu einem Cluster. Dies wird fortgesetzt, bis die gewünschte Anzahl von Clustern erreicht ist.

Für große Mengen D ist diese Vorgehensweise natürlich zu aufwendig; hier gibt es effizientere Alternativen, die man in der einschlägigen Literatur findet.

b) Lineare Diskriminanzanalyse

c) Support Vector Machines

Nicht alle Datensätze können durch eine lineare Diskriminanzfunktion getrennt werden. Zerfällt die Datenmenge $D \subset \mathbb{R}^n$ etwa in zwei Komponenten D_1 und D_2, wobei die Punkte aus D_1 nahe beim Nullpunkt liegen, die aus D_2 aber in weiter Entfernung, so wird es üblicherweise keine solcheFunktion geben. Betten wir aber D ein in $\mathbb{R}^{n+1}$, indem wir für jeden Punkt als $(n + 1)$-te Komponente seinen Abstand zum Nullpunkt hinzunehmen, ist die Klassifikation nach Abstand über eine einfache lineare Funktion möglich.

Die Grundidee bei *support vector machines* beruht darauf, einen gegebenen Datensatz $D \subset \mathbb{R}^n$ so in einen höherdimensionalen Raum $\mathbb{R}^m$ einzubetten, daß die interessierenden Komponenten dort durch lineare

Funktionen getrennt werden können. Die Dimension m kann dabei teilweise beträchtlich größer als n sein; teilweise arbeitet man sogar mit unendlichdimensionalen Räumen.

Um festzustellen, auf welcher Seite einer gegebenen Hyperebenen im $\mathbb{R}^m$ ein Punkt liegt, müssen wir Skalarprodukte im $\mathbb{R}^m$ berechnen, was für große Werte von m schnell aufwendig wird. Daher muß die Einbettung $\mathbb{R}^n \to \mathbb{R}^m$ so gewählt werden, daß das Skalarprodukt der Bilder zweier Vektoren $u, v \in \mathbb{R}^n$ im $\mathbb{R}^n$ durch eine möglichst einfache Funktion $k(u, v)$ auf $\mathbb{R}^n \times \mathbb{R}^n$ berechnet werden kann. Eine solche Funktion bezeichnet man als einen *Kern,* und meist geht man aus von einem Kern, um dann eine Einbettung zu konstruieren, bezüglicher derer er das Skalarprodukt der Bildvektoren berechnet.