

die über die spezifische Anordnung kleiner kaskadischer Einheiten im Top-down-Verfahren Fehler korrigieren und diese Behebung in das nächste System weitertragen, die also das »eigenständige« Lernen neuronaler Netzwerke erklären soll.

Die Mathematisierung von Wahrnehmung drückt die allumfassende Implementierung mathematischer Logiken und Prämissen in heutige erkenntnisleitende Werkzeuge, Theorien und Modelle sowie die verwendeten Technologien aus. Die Theorien und Methoden sind durchdrungen von mathematischen Logiken, rational-logischen Abläufen und statistischen Vorhersagen. Im Folgenden werden die Effekte der Mathematisierung am Beispiel des algorithmischen und des vorhersagenden Gehirns vorgestellt.

3 Etwas Besseres als die Natur? Effekte der Mathematisierung von Wahrnehmung: *algorithmic* und *predictive brain*

Computational Neurosciences sind nicht nur ein erstaunlich interdisziplinär arbeitendes Feld, auch die Bezüge in andere Teilbereiche der Hirnforschung treten deutlich zutage. Allianzen und Überschneidungen gibt es nicht nur mit experimenteller Mathematik, Informatik, theoretischer Physik, Neuroinformatik und künstlicher Intelligenzforschung, die Debatten über die physische Verfasstheit des Gehirns reichen auch in die Psychologie, die Philosophie (des Geistes) und die Kognitionswissenschaften. Aus einigen Überschneidungen der Teilbereiche sind wiederum neue Subdisziplinen entstanden. Die Neurowissenschaft konnte sich so einerseits zu einer eigenständigen Disziplin herausbilden, die sich mit dem Gehirn beschäftigt, gleichzeitig deckt die Bezeichnung ein enorm breites Feld mit vielen Unterdisziplinen und methodischen Zugängen ab, die teilweise ein ähnliches, aber mitunter auch völlig unterschiedliches Verständnis des Gehirns, des Mentalen, der Psyche und der jeweiligen zerebralen Prozesse beinhalten. Der Fokus dieses Buches liegt nur auf einem sehr kleinen Bereich dieses breiten Feldes, auch, und das wurde auf den letzten Seiten hoffentlich deutlich, wenn es immer wieder konzeptionelle wie theoretische Überschneidungen gibt und epochenabhängige Tendenzen, in denen sich eine mathematische Mechanik des Geistes durchsetzte.

Hinter die heute wohl unumstrittene »Einsicht, dass niemand ohne Gehirn denken könne« (Fitsch 2013, 5), dass es also das Gehirn ist, das Gedanken, Gefühle, Bewusstsein und Erinnerung hervorbringt, kann nicht mehr zurückgegangen werden. Ebenso hat sich die Erkenntnis durchgesetzt, dass für die

Verarbeitungs- und Wahrnehmungsprozesse ein Zusammenspiel neuronaler Netzwerke verantwortlich ist. Wie diese Architektur aussehen muss, um ein komplexes, neuronales System mit diesen vielfältigen Eigenschaften hervorzubringen, das ist heute Ausgangspunkt sowohl vieler gesellschaftlicher Debatten als auch für die verschiedenen Subdisziplinen in den Neurowissenschaften.

In den Computational Neurosciences werden Neuronale Netzwerke unter der Annahme eines synaptischen Verhaltens einzelner Neuronen konzeptualisiert, deren Tätigkeit auf die Feuerungsraten einer ganzen Neuronengruppe übertragen wird. Auf Grundlage der singulären Feuerungsrate wird also das Zusammenwirken neuronaler Gruppen modelliert und simuliert. Zentral hierfür ist die Annahme von Verbindungsstärken, die sich aufgrund von Erfahrung und Nutzung als Lerneffekt verdicken und bei häufigerer ›Verwendung‹ ausprägen:

As we are sure that many synapses are plastic, that is, they can change their strength with experience, it is also important to know the abstract rules imposed in such changes by the intricate biochemistry of a neuron. Thus a net is characterized by the properties of the units that make it up, the way they are connected together, and the algorithms used to change the strength of those connections. (Crick 1989, 129)

Völlig unklar hinsichtlich der Ausgestaltung dieser neuronalen Architektur ist jedoch unter anderem, welche Kriterien für die Entwicklung neuronaler Strukturen verantwortlich sind. Wie führt die Feuerungsrate einer einzelnen Synapse zu synaptischen Verbindungen, die sich wiederum zu neuronalen Clustern zusammensetzen? Welche Merkmale entscheiden darüber, welches der strukturell voneinander abgegrenzten Netzwerke sich gegen die umliegenden neuronalen Verbindungen durchsetzt? Neben anderen Erklärungsansätzen in der Entwicklungstheorie des Nervensystems identifiziert ein Ansatz den Wettbewerb und Konkurrenzverhalten als wichtiges Grundprinzip in der plastisch organisierten Ausbildung von Nervenverbindungen: »Plasticity through competition [...] the labels do not change, and the flexibility in connectivity is possible through competition.« (Sterrat et al. 2014, 302)

Die hier vorgestellten und auf Analogien aus der Computertechnologie basierenden mathematischen Logiken konnten mithilfe rechenstarker Computer komplexer ausgestaltet werden. Die Basis dieser Überlegungen und der komplexer werdenden Modelle sind und bleiben jedoch die vorher vorgenom-

menen Reduktionen, die somit immer noch Teil des gegenwärtigen Verständnisses unseres Gehirns sind.

Heute sind Algorithmen unter anderem ein mathematisches Werkzeug der Beweisführung. Algorithmen sind auf Erfahrung beruhende, operationalisierte Verfahren, um Probleme zu lösen und/oder Handlungsanweisungen zu erstellen und an die Hand zu geben. In stochastischen Algorithmen, die Wahrscheinlichkeitsrechnung und Statistik verbinden, werden Problemlösungen und Handlungsanweisungen ein und dasselbe. Um dies zu verdeutlichen, stelle ich im Folgenden zwei Beispiele vor: das *algorithmic brain* oder auch algorithmische Gehirn und das Konzept des *predictive brain* oder auch vorhersagendes Gehirn.

Algorithmisches Gehirn

Was ist mit dem algorithmischen, was mit dem vorhersagenden Gehirn gemeint? Ein Algorithmus beschreibt zunächst einmal die Vorgehensweise bei der Lösung eines Problems. Etwas konkreter charakterisiert ein Algorithmus die Art der Vorgehensweise, mit der ein Problem gelöst werden soll, nämlich das schrittweise Abarbeiten klar vorgegebener, endlicher Arbeitsaufträge in einer durchführbaren, bestimmten Zeitspanne (*finite time*). Algorithmen gibt es keineswegs erst seit der Erfindung von Computern. Die Bezeichnung Algorithmus geht auf den Namen des in Bagdad lehrenden Mathematikers, Astronomen und Geografen Abu Dscha'far Muhammad ibn Musa Chwārizmī (780–835/850) zurück. Dieser beschäftigte sich mit indischen Zahlschriften, übertrug die Null aus dem indischen in das arabische Zahlensystem und machte die Null damit für heutige ›moderne‹ mathematische Zahlensysteme nutzbar. Eben jene Schrift, in der er die Null in die arabische Zahlschrift einführte, trug in einer lateinischen Fassung den Titel *Algoritmi de numero Indorum*, also »Al-Chwarizmi über die indischen Zahlen« (herausgegeben in Rom 1857), von dem später die Bezeichnung Algorithmus abgeleitet wird. Chwārizmī gilt bis heute als einer der bedeutendsten Mathematiker, neben der Übertragung der Null aus dem indischen Zahlensystem sowie der Einführung konkreter Regeln und eindeutiger Handlungsvorschriften für das Rechnen mit Dezimalzahlen geht auch der Begriff der Algebra, also das Rechnen mit Unbekannten in Gleichungen, auf seine Schriften zurück. Damit schuf er mehrere wichtige mathematische Werkzeuge, die bis heute, nicht zuletzt im Computer, Anwendung finden. Auch George Boole lässt sich viele Jahre später in seinen Arbeiten zur boolschen Algebra von der indischen Logik und dem

indischen Zahlensystem inspirieren (s. Kap. 2). Seine Algebraisierung der Logik gilt heute ebenfalls als Vorläufer der Computerlogik. Algorithmen wurden also zum Lösen mathematischer Probleme eingeführt und sind seither Werkzeuge zum Rechnen. Für die Implementierung in den Computer erhielten Algorithmen eine weitere formalisierende Zuspitzung, um sie für die Logik des Computers nutzbar zu machen.

Die Funktionsweise von Algorithmen wird gerne mit Kochrezepten verglichen, als detaillierte Beschreibung exakter Schritte, die am Ende ein immer gleich schmeckendes Gericht hervorbringen. Dieser Vergleich ist irreführend und unzureichend und verkennt das Wesentliche eines Algorithmus.

Bei Algorithmen geht es nicht um den einen Plan, um aus einem Labyrinth zu entkommen. Es geht darum, wie man den Weg immer findet. Darin liegt ihre Stärke, und das macht sie verdächtig. Der Unterschied zwischen einem Kochrezept, das alles auf einen Einheitsbrei reduziert, und dem Algorithmus, bei dem aus einem Prinzip eine ungeahnte Vielfalt wächst, ist leicht zu übersehen. (Stiller 2015, 63)

Der Mathematiker Sebastian Stiller sieht die Aufgabe von Algorithmen eher darin, das Verhältnis von Einzelfall und Prinzip neu zu ordnen:

Oft geht es nicht um die Verhandlungen für ein einzelnes Geschäft, sondern darum, wie man unzählige unterschiedliche Geschäfte immer wieder erfolgreich abschließt. Nicht um den einen Impfstoff, sondern um das Verfahren, mit den immer neuen Grippeviren Schritt zu halten. Nicht um den von Hand maßgeschneiderten Anzug, sondern die Konfektion, nicht um die individuelle Empfehlung des Buchhändlers, sondern die personalisierten Filmempfehlungen von Online-Filmverleihen, nicht um das persönliche Verkaufsgespräch, sondern die personalisierten Werbebanner im Internet, nicht um die Eroberung der einen Liebe des Lebens, sondern um die Masche des Pick-Up-Artist. (Ebd.)

Die Bedeutsamkeit von Algorithmen liegt also laut Stiller darin, dass sie, einmal gefunden, allgemeingültige Lösungswege anbieten. Dank schneller Rechenleistung bedeute dies aber gleichzeitig nicht länger, dass es nur ein Entweder-oder gibt, »Einzelfall oder Einheitsbrei. Es gibt etwas dazwischen: das einfache Prinzip, das Vielfalt entstehen lässt.« (Ebd., 64) Dieses Prinzip liegt in den Anfangskonfigurationen der Lösungswege; das Wesentliche eines Algorithmus aber ist zunächst »nicht die feste Regel, sondern die Vielfalt dessen, was sich daraus ergibt. Ein Kochrezept produziert immer das gleiche

Gericht. Ein Algorithmus produziert je nach Input eine unvorhersehbare Vielfalt.« (Ebd., 65) Eine Vielfalt, die jedoch stark auf der Verallgemeinerung der Universalisierung von Problemen und Lösungsansätzen beruht und die eine Einheit der Klassifikationen sowie standardisierte Situationen voraussetzt. Denn letztlich bleiben Algorithmen mathematische Operationen in kleinen Schritten, die sich über ihre klassifikatorische Familie, der sie zuzuordnen sind, bestimmen.

Die Theorie der Algorithmen entstammt der gleichen Ideenfamilie wie die Bemühungen um die Grundlagen der Arithmetik, allgemeiner die Grundlagen der Mathematik, Ende des 19. und zu Beginn des 20. Jahrhunderts. Dabei geht es darum, jenen Teil des Selbstverständnisses der Mathematik einzulösen, nach dem diese aus elementaren und formalisierbaren Schlussfolgerungen aufgebaut werden. (Ebd., 49)

Dieses algorithmische Denken, betont Stiller, verändert einerseits *was* wir denken, aber auch *wie* wir denken: »Wir denken mit anderen Methoden und Kriterien über diese Welt nach. [...] Es verändert den Einzelnen und die Gesellschaft im Ganzen.« (Ebd., 32) Stiller geht noch einen Schritt weiter, bewertet diese Veränderung und sieht unsere »alltägliche Klugheit« jetzt schon durch »die Produkte des algorithmischen Denkens verkümmern.« (Ebd.)

In den Neurowissenschaften stellt sich nunmehr seit einigen Jahren die Frage, ob das menschliche Gehirn über Algorithmen verfügt. Mit Algorithmus kann dabei zweierlei gemeint sein: erstens auf neuronaler Verarbeitungsebene, also mit Blick darauf, ob die Verarbeitung in den neuronalen Netzwerken anhand algorithmischer Handlungsabläufe zu Entscheidungen führt, oder zweitens, ob das menschliche Denken und Handeln sich *per se* algorithmischer Logiken bedient. Für beide Verständnisse werde ich im Folgenden ein Beispiel vorstellen.

»Does the human brain have algorithms?«, fragen die beiden Autor*innen Sule Yildirim (heute Yildirim Yayilgan) und Ronald L. Beachell (2006) und meinen damit Algorithmen, die klare Schritt-für-Schritt-Anweisungen beinhalten, Problemlösungsansätze beschreiben oder Lösungen ausführen (vgl. ebd., 1). Ausgehend von dieser Definition, stellt sich nun die Frage, ob es einen so verstandenen Algorithmus auch im Gehirn gibt:

Algorithms in the context of our approach consist of a series of steps and/or rules to perform a task. The non-symbolic algorithms may be innate or learned. Some combinations of steps and rules may exist in the brain as

building block algorithms which may be combined with other steps and rules to form complex algorithms. (Ebd.)

Und weiter definieren Yildirim Yayilgan und Beachell »non-symbolic concepts as a means to accomplish some tasks and therefore to explain some functions of cognition« (ebd., 7). Im Grunde genommen verstehen sie Algorithmen als Prozesse, die im Sinne von Wenn-dann-Gesetzen in komplexen Systemen ablaufen. Als Beispiel greifen sie auf eine Situation zurück, in der das Gehirn eine Entscheidung darüber treffen soll, ob das Seewasser zum Baden (zu) kalt ist und wenn ja, was dann zu tun sei:

The biological neural realization that we choose for the human brain has representations for the concept of cold lake water. It also has representations for steps (processes or actions) and rules including »sensing« – »If-Then« and »get out of«. One can link these to each other as follows in a biological neural realization where the concepts are represented by neuron clusters and processes (steps or rules in an algorithm) are represented by neural structures. (Ebd., 4)

Um die Anwendung eines algorithmisch arbeitenden Gehirns zu garantieren, braucht es klare Aussagen, die wiederum klare Anweisungen ergeben. Der Algorithmus für das Verhalten in kaltem Wasser kann als Text oder im Anschluss an die Programmiersprache als Befehl ausgedrückt werden: »Wenn das Wasser zu kalt ist (definiert durch eine festgelegte Temperatur), dann geh aus dem kalten Wasser.« – »This can be broken down and represented symbolically as a sequence of steps: sense (LAKEWATER) IF cold, THEN (next step) get out of (LAKEWATER).« (Ebd.) Die beiden Autor*innen kommen zu dem Ergebnis, dass es sehr gut sein kann, dass das Gehirn mithilfe algorithmischer Formalisierungen Reize verarbeitet und Entscheidungen trifft.

Dieses erste Beispiel zeigt, wie die Logik parallel verarbeitender Computer und die Notwendigkeit, diese in die Problematik der Wenn-dann-Verschaltungen einzubauen, in die Vorstellung von Denkprozessen Eingang fand. Das im Folgenden präsentierte zweite Beispiel zeigt, wie die Logik der großen Zahl und der Wahrscheinlichkeit in die Vorstellung »geregelter Abläufe menschlichen Lebens integriert wurde.

Algorithmen sind klar überprüfbare Handlungsanweisungen, deren Ergebnisse weder der Wahrheitsfindung dienen noch eine Grundlage bieten, um ethische und politische Fragen zu diskutieren. Algorithmen, so die Überzeugung zumindest der beiden Autoren Brian Christian und Tom Griffiths im

zweiten hier aufgeführten Beispiel, lassen sich nicht nur in die Verarbeitung paralleler Prozesse einbinden. Der Mehrwert dieser Art der Entscheidungsfindung ergibt sich aus der konsistenten Anbindung an die Naturgesetze der Wahrscheinlichkeitsrechnung, die nicht nur die kognitiven Prozesse bestimmen, sondern das Denken im Allgemeinen:

Most broadly, looking through the lens of computer science can teach us about the nature of the human mind, the meaning of rationality, and the oldest question of all: how to live. Examining cognition as a means of solving the fundamentally computational problems posed by our environment can utterly change the way we think about human rationality. (Christian/Griffiths 2016, 4)

So lässt sich in der Kognitionswissenschaft auch eine Algorithmisierung der Logik als erkenntnistheoretische Brille zur Erfassung der Welt ausmachen. Es zeigt ein weiteres Mal den Wandel, auf den bereits an anderer Stelle verwiesen wurde (die ›fehlende‹ halbe Sekunde in Kap. 4): der Wandel von der Auseinandersetzung und Nutzbarmachung des Unbewussten zur Oberhoheit und Nutzbarmachung des Bewussten. Im 2016 erschienenen Buch *Algorithms to live by* geben Christian und Griffith ein wirkmächtiges Credo aus, das sich als ein Ergebnis einer Mathematisierung der Wahrnehmung bezeichnen lässt: Den Autoren zufolge laufen alle Entscheidungsfindungen des Menschen nach algorithmischen Regeln ab. Sie halten fest: »Living by the wisdom of computer science doesn't sound so bad after all. And unlike most advice, it's backed up by proofs.« (Ebd., 9)

Welcher Art sind die Beweise, die die Informatik bereithält und auf die hier verwiesen wird? Christian und Griffiths argumentieren, dass sich Algorithmen auf logischem Wege, quasi als goldene Regeln, in unserer Art des Denkens oder konkreter in der menschlichen Entscheidungsfindung wiederfinden lassen. Als Beispiele verweisen die beiden auf klassische Fragen aus der Statistik, übertragen auf alltägliche, aber weitreichende Entscheidungen wie die Partner*innenwahl oder der Aktienkauf. Sie definieren in ihrer Theorie Algorithmen als Abfolge festgelegter Schritte zur Lösung eines Problems. Damit werden Algorithmen zu Anleitungen, die ›uns‹ das Leben erleichtern: »[A]n algorithm is just a finite sequence of steps used to solve a problem, and algorithms are much broader – and older by far – than the computer. Long before algorithms were ever used by machines, they were used by people.« (2016, 6) Um das algorithmische Konzept von der mechanisch-tech-

nischen Logik des Computers zu lösen, erweitern sie den Algorithmusbegriff kurzerhand um alle Tätigkeiten, für die es einen Ablaufplan gibt:

When you cook bread from a recipe, you're following an algorithm. When you knit a sweater from a pattern, you're following an algorithm. When you put a sharp edge on a piece of flint by executing a precise sequence of strikes with the end of an antler – a key step in making fine stone tools – you're following an algorithm. Algorithms have been a part of human technology ever since the Stone Age. (Ebd.)

Diese sehr allgemein gehaltene Definition von ›Algorithmen fürs Leben‹ (*algorithms to live by*) basiert auf lebenspraktischen Weisheiten, auf der praktischen Anleitung menschlicher Angelegenheiten und ist eher eine sehr menschliche ›Art des Denkens‹ als ein feststehender Wissensschatz. Durch die Verortung von algorithmischen Abläufen in Tätigkeiten, die bereits im Steinzeitalter vom Menschen durchgeführt wurden, wird diese ›Art des Denkens‹ naturalisiert und der kulturelle und erkenntnistheoretische Aspekt von Algorithmen negiert. Durch die Übertragung algorithmischen Denkens (das eigentlich ein Tun ist) bereits auf das Steinzeitalter und auf alle Formen alltäglichen Handelns (wobei außer Acht gelassen wird, dass es sich beim Kochen und Stricken um Kulturpraktiken handelt) können die beiden Autoren Algorithmen als intuitive, dem Menschen inhärente Weise des Schlüsseleinhens formalisieren und gleichzeitig, wenn auch unausgesprochen, in die menschliche DNA einbauen. So steht bei Christian und Griffiths die Suche nach einer allgemeingültigen Formel, nach der ›Logik der Logik‹ im Vordergrund, die mit mathematischen Gesetzmäßigkeiten plausibilisiert werden soll. »In this book, we explore the idea of human algorithm design – searching for better solutions to the challenges people encounter every day.« (Ebd., 6) Das Studium des Innenlebens von Computern, so die Autoren, kann darüber Aufschluss geben, wie wir denken und entscheiden, was wir glauben und wie wir uns verhalten sollten: »Even in cases where life is too messy for us to expect a strict numerical analysis or a ready answer, using intuitions and concepts honed on the simpler forms of these problems offers us a way to understand the key issues and make progress.« (Ebd., 7)

Algorithmen sind ein altbewährtes mathematisches Werkzeug und ihr Einsatz ist nicht auf Computer beschränkt. Dennoch sind sie mathematische Anleitungen für mathematische Probleme, deren intrinsisch eingelagerte algorithmische Mehrwerte durch ihren Einsatz in mathematisch-technisch engineerten Welten des Computers einen weiteren erkenntnistheoretischen

Wandel erfahren haben. Christian und Griffiths haben entschieden, diesen historischen, epistemischen Bedingungen keine weitere Beachtung zu schenken, sondern Algorithmen als Metapher für ein tieferes Verständnis der Entscheidungsprozesse im menschlichen Denken anzunehmen:

The solutions to everyday problems that come from computer science tell a different story about the human mind. Life is full of problems that are, quite simply, hard. And the mistakes made by people often say more about the intrinsic difficulties of the problem than about the fallibility of human brains. Thinking algorithmically about the world, learning about the fundamental structures of the problems we face and about the properties of their solutions, can help us see how good we actually are, and better understand the errors that we make. (Ebd., 8)

Die 37-Prozent-Regel als Algorithmus für alltägliche Entscheidungen

Ein Beispiel einer ›algorithmischen Lebensweisheit‹, die laut Christian und Griffiths Aufschluss über die menschliche algorithmische Verfasstheit geben soll, ist die sogenannte 37-Prozent-Regel. Sie gilt als Antwort auf das *optimal stopping*-Problem, also die Frage, wann man aufhören sollte weiterzusuchen. Dieses Problem existiert auch in der mathematischen Berechnung selbstorganisierender, komplexer Systeme, die Daten und Prozesse iterierend (wiederholend) in das System/Netzwerk einspeisen. Um diesen Iterationsmechanismus zu stoppen, braucht es eine Variable, die extra in das System programmiert werden muss. Und selbstverständlich stellt sich die Frage, wann eine Berechnung, und damit das Wachstum eines Systems, gestoppt werden muss. Mit der goldenen Regel von 37 Prozent können laut Christian und Griffiths das sogenannte Sekretär*innenproblem ebenso wie Ungereimtheiten in Liebesangelegenheiten ein für alle Mal gelöst werden.

Um diese algorithmische Regel anzuwenden, braucht es zunächst eine konkrete Problembeschreibung, die sich nicht darüber bestimmt, welche Option am Ende wirklich gewählt wird, sondern wie viele Optionen überhaupt in Betracht gezogen werden, um einen Pool an Möglichkeiten zu generieren. Begibt man sich auf die Suche nach einer*m passenden Sekretär*in oder dem perfekten Liebesmatch, bestimmt sich das Problem der Suche nur noch durch zwei Variablen: zu früh oder zu spät aufzuhören. Hört man zu früh auf, bleibt die am besten geeignete Person unentdeckt. Hört man zu spät auf, werden

die Ansprüche zu hoch gesteckt, und es wird auf eine noch bessere Bewerbung gewartet, die gar nicht kommen kann. Die optimale Strategie besteht also darin, die richtige Ausgewogenheit zwischen zu früh und zu spät zu finden, ein Zustand, der sich numerisch exakt mit 37 Prozent beziffern lassen soll:

Enumerating these scenarios for four applicants tells us that we should still begin to leap as soon as the second applicant; with five applicants in the pool, we shouldn't leap before the third. As the applicant pool grows, the exact place to draw the line between looking and leaping settles to 37 % of the pool, yielding the 37 % Rule: look at the first 37% of the applicants, choosing none, then be ready to leap for anyone better than all those you've seen so far. (Ebd., 16)

Aber wann sind 37 Prozent erreicht, und wann sollte man aufhören zu suchen? Das sogenannte Sekretär*innenproblem lässt sich ebenso auf andere große Fragen des Lebens, zum Beispiel die Suche einer*s Partner*in fürs Leben, anwenden. In Christians und Griffiths' Buch stellt sich ein Mathematiker dieser Aufgabe:

He didn't know how many women he could expect to meet in his lifetime, but there's a certain flexibility in the 37 % Rule: it can be applied to either the number of applicants or the time over which one is searching. Assuming that his search would run from ages eighteen to forty, the 37 % Rule gave age 26.1 years as the point at which to switch from looking to leaping. A number that, as it happened, was exactly [the persons] age at the time. So when he found a woman who was a better match than all those he had dated so far, he knew exactly what to do. He leapt. I didn't know if she was perfect (the assumptions of the model don't allow me to determine that), but there was no doubt that she met the qualifications for this step of the algorithm. So I proposed, he writes. (Ebd., 18)

Ganz so einfach läuft es in Liebesdingen, wenn man die Partner*innensuche bei Christian und Griffiths denn so nennen möchte, natürlich nicht immer ab, selbst bei den größten Mathematiker*innen nicht. Dass sich die goldene Regel der 37 Prozent dennoch immer wieder, zumindest in der Retrospektive, bestätigt, das zeigt sich auch bei dem Astronomen und Mathematiker Johannes Kepler, der, so die beiden Autoren, nicht nur mit den Sternen kämpfte, sondern auch irdische Sorgen bewältigen musste:

After the death of his first wife in 1611, Kepler embarked on a long and arduous quest to remarry, ultimately courting a total of eleven women. Of the first four, Kepler liked the fourth the best (»because of her tall build and athletic body«) but did not cease his search. »It would have been settled,« Kepler wrote, »had not both love and reason forced a fifth woman on me. This one won me over with love, humble loyalty, economy of household, diligence, and the love she gave the stepchildren.« [...] His thoughts remained with number five. After eleven courtships in total, he decided he would search no further. »While preparing to travel to Regensburg, I returned to the fifth woman, declared myself, and was accepted. (Ebd., 19)

Kepler und seine fünfte Frau Susanna Reuttinger, über deren Leben man in der mathematischen Denkübung leider nichts weiter erfährt, lebten, zumindest nach den Standards der Autoren zu urteilen, glücklich bis ans Ende ihrer Tage.

Die von Christian und Griffiths aufgeführten Beispiele zeigen, dass die Anwendung der algorithmischen Logik auf einer klaren Trennung von einerseits einer numerisch getriebenen Ratio und andererseits von in der Gefühlswelt verorteten Fragen wie dem Finden der großen Liebe beruht. Erst die Unterwerfung der Gefühlsebene unter die Logik von Zahlen, Prozenten und Statistiken führt zu einer Lösung zum vermeintlichen Wohle aller. Hierfür muss zunächst der zu erfassende Gegenstand als klar definierbares (da erst dann lösbares) Problem deklariert werden. Die Rahmung als Problem bestimmt die Art, wie damit umgegangen werden muss: Allein durch die Einhaltung einer bestimmten Handlungsweise kann die Aufgabe zu einem erfolgreichen Abschluss geführt werden. Das Problem wie die Problemlösung basieren auf den Alltagserfahrungen der beiden weißen, männlichen, akademischen Autoren, die die Trennung von rationaler und emotionaler Ebene durch die nicht kontextualisierte Verwendung undefinierter Begrifflichkeiten wie Liebe, Anziehung und Zuneigung statistisch einhegen und somit der Logik eines verallgemeinerbaren Algorithmus unterwerfen. Die 37-Prozent-Regel ist das Stichwort, die Regel, die besagt, wann man aufhören sollte zu suchen, und die Aufrichterfordern, nicht immer alle (möglichen) Optionen in die Entscheidungsfindung mit einzubeziehen. Das algorithmische Gehirn zeichnet sich, so Christian und Griffiths, durch die unbewusste, aber stete Anwendung statistischer Regeln aus. Das Beispiel der 37-Prozent-Regel wurde hier angeführt, um zu illustrieren, wie verbreitet die Übertragung mathematischer Modelle auf das Gehirn ist.