Einleitung

Die Biologie mit all ihren vielfältigen Teilgebieten wie Botanik, Zell- und Molekularbiologie, Genetik oder Humanbiologie ist als ein reges Forschungsfeld in stetem Wandel. Sie ist gleichzeitig, mehr als andere Domänen, ständigen Wechselwirkungen mit angrenzenden Bereichen wie Chemie, Physik oder Medizin und seit einigen Jahrzehnten auch technischeren Bereichen wie der Informatik unterworfen. Gerade durch die Interaktion mit der Informatik hat sich ein eigenes Fachgebiet entwickelt, die Bioinformatik. In der Praxis führt das dazu, dass bei vielen Naturwissenschaftlern inzwischen weitreichende Programmiererfahrungen vorhanden sind, die deutlich über Tabellenkalkulationsprogramme hinausgehen und nicht nur die Thematik der Handhabung wissenschaftlicher Daten umfasst.

Zugleich hat sich in den letzten Jahren die Biologie gemeinsam mit der Medizin, der Chemie sowie den Pflege- und den Agrarwissenschaften zu einem spannenden, interdisziplinären Feld entwickelt: den Lebenswissenschaften (engl. *Life Sciences*). In diesem Gebiet fließen die Grenzen der »klassischen« Disziplinen zusehends ineinander. Ein Buch, das sich lediglich mit der Schnittmenge zwischen Biologie und Informatik beschäftigt, erscheint uns daher nicht weitreichend genug. Wir haben uns das Ziel gesteckt, den Bogen hier wesentlich weiter zu spannen.

Dementsprechend geht es in diesem Buch primär darum, angewandte Probleme der Lebenswissenschaften mithilfe der Informatik zu lösen. Es wird dabei nur so weit auf biologische oder medizinische Hintergründe eingegangen, wie es für das Verständnis des Buchs nötig ist. Das Vorgehen wird mit viel Beispielcode, teilweise im Stil eines Kochbuchs, dargestellt. Wann immer möglich, sollen Übungsaufgaben und Praxisprojekte tiefer in die Materie führen.

In diesem Buch werden vor allem folgende allgemeine Themenbereiche behandelt:

- *Big Data*: die Verarbeitung einer großen Menge sich ständig ändernder Daten.
- Das Heranziehen und Adaptieren effizienter, robuster und zuverlässiger Lösungen (insbesondere *Heuristiken* und *Algorithmen*) für bestehende Probleme.
- Informationen über *State-of-the-Art-Lösungen*, *-Bibliotheken* und *-Algorithmen*.

Somit werden alle wichtigen Grundlagen für Studierende und Praktiker in den Lebenswissenschaften, der Biologie, der Bioinformatik und anderen Fächern behandelt: Wie werden Daten und Informationen verwaltet (sogenanntes *Data Management*), wie werden sie über Algorithmen und externe Bibliotheken verarbeitet, und wie werden damit schlussendlich Probleme im Sinne der zugrunde liegenden wissenschaftlichen Fragestellung gelöst? Unser Ziel ist es, Ihnen eine Vorstellung von den verfügbaren und etablierten Softwarelösungen und Konzepten sowie von generellen Herangehensweisen bei datenbasierten Problemstellungen zu vermitteln. Nicht zuletzt legen wir Wert auf Verweise darauf, wo Sie auch über dieses Buch hinaus weitere Hilfe finden können.

Entwickelt hat sich das vorliegende Buch aus zahlreichen Lehrveranstaltungen, die die Autoren an den Universitäten Bonn, Köln und München über die vergangenen Jahre hinweg angeboten haben; hinzu kommen die eigenen Erfahrungen aus der fachlichen und beruflichen Praxis. Wir haben während der Ausarbeitung und schriftlichen Niederlegung der Kursmanuskripte von der interdisziplinären Arbeit zwischen Mathematik, Informatik, Biologie und den weiteren Lebenswissenschaften stark profitiert und hoffen, dies an Sie weitergeben zu können. Unser Dank gilt auch den Generationen von Studenten, die durch ihre Rückfragen und Rückmeldungen die Darstellung geschärft haben und dadurch helfen konnten, den Fokus dieses Buchs entsprechend auszurichten.

Bei der Themenauswahl haben wir uns von drei Faktoren leiten lassen: Welche Themen sind derzeit wichtig und emergent? Welche Themen sind bereits langfristig in Erscheinung getreten und bleiben somit auch noch für mindestens eine weitere Generation von Bedeutung? Und zu guter Letzt haben uns natürlich auch unsere persönlichen Präferenzen und Interessen geleitet. Mehr zu den einzelnen Themen berichten wir, wenn wir den Aufbau des Buchs beschreiben.

Java in den Lebenswissenschaften

Es muss auch erwähnt werden, dass die Lebenswissenschaften ein sehr agiles Feld sind und es fast täglich zu Neuentwicklungen in den verschiedenen thematischen Bereichen wie z.B. Assays, Datentypen, Dateiformaten und Software kommt. Doch bleiben grundlegende Konzepte weitestgehend davon unberührt, und auch die Trends zur Verwendung neuer Programmiersprachen verlaufen eher langsamer. Daher erscheint es uns nach wie vor sinnvoll und wichtig, Java als Grundlage für dieses Buch anzunehmen. Java erblickte 1995 das Licht der Welt und verfügt damit über eine große Anzahl von Bibliotheken und eine große Entwicklergemeinde. Darüber hinaus zeichnet sich Java durch die hohe Prozessierungsgeschwindigkeit und seine Portabilität aus; es kann auf allen gängigen Systemen ohne Probleme ausgeführt werden.

Auch wenn der größte Teil wissenschaftlicher Infrastruktur auf Unix/Linux aufbaut, können daher die Beispiele aus diesem Buch ebenso auf Mac- oder Windows-

Systemen ausgeführt und weitestgehend problemlos übernommen werden. Sollte es hier Besonderheiten geben, weisen wir an Ort und Stelle darauf hin. Die Ausnahme bildet der immer wieder vorkommende Hinweis auf BASH, insbesondere wenn es um HPC-Umgebungen (High-Performance Computing) geht. Diese Shell lässt sich unseres Wissens nur auf einer ganz kleinen Anzahl von Windows-Systemen nutzen. Auch verweisen wir bei grafischen Installationsroutinen unter Windows grundsätzlich auf andere Quellen. An manchen Stellen gehen wir also implizit von einer Unix-artigen Umgebung aus.

Kritiker mögen bemängeln, dass doch gerade in den Lebenswissenschaften Python (entstand 1991) oder Matlab (das auf die 1970er-Jahre zurückgeht) benutzt würde. Dem möchten wir allerdings entgegensetzen, dass dies schon beim Blick auf eine andere Forschungseinrichtung, in eine andere Arbeitsgruppe oder bereits bei einer konkreten Aufgabe nicht mehr der Fall sein muss. Im Gegensatz zu den oben genannten Programmiersprachen ist Java schlicht schon länger in manchen Bereichen präsent und etabliert – selbst in professionellen Umgebungen weit außerhalb der Wissenschaft. Ebenso wird es in vielen Open-Source-Projekten verwendet: Ein nicht unerheblicher Teil der verfügbaren wissenschaftlichen Software ist in Java entwickelt worden. Prominente Beispiele sind etwa das GATK-Toolkit zur Analyse von DNA-Sequenzen und die Bildverarbeitungsumgebung ImageJ. Letztlich bleibt bei dieser Fragestellung aber immer eine gewisse Ambivalenz, und deswegen blicken wir an manchen Stellen auch über Java hinaus, z. B. auf BASH.

Prinzipiell bietet sich Java auch aufgrund der Vielzahl hochqualitativer kostenloser Tutorials für einen Start in die Programmierung und speziell in die Objektorientierung an. Wir setzen daher explizit einfache Programmierkenntnisse voraus – mehr dazu im folgenden Abschnitt. Die Vorkenntnisse, sofern sie nicht in Java vorliegen, können sehr leicht auf Java übertragen werden, und auch umgekehrt profitieren Sie von den neuen Erkenntnissen aus diesem Buch, wenn Sie in anderen Sprachen programmieren. Probleme, die nur in einer speziellen Sprache gelöst werden können, gibt es heutzutage nur noch sehr selten – aber es gibt viele Probleme, die sich in Java besonders einfach lösen lassen.

Zielgruppe und Voraussetzungen dieses Buchs

Zielgruppe dieses Buchs sind Studierende, Berufstätige und Dozenten der Biologie, der Bioinformatik, der Informatik, der Life Sciences und allen verwandten Naturwissenschaften. Sie sollten Grundkenntnisse im Bereich der Biologie und Lebenswissenschaften sowie Kenntnisse mindestens einer höheren Programmiersprache mitbringen – oder zumindest die Bereitschaft, sich entsprechend einzulesen.

Wenn Sie Grundkenntnisse in der Algorithmik mitbringen, werden Sie diesem Buch schneller folgen können. Wir setzen dies aber bewusst nicht voraus, stattdessen verweisen wir stets auf die Grundlagenkapitel in diesem Buch und beschreiben – wann immer es nötig ist – die Algorithmen und Herangehensweisen. Der Leser

soll zu einem selbstständigen Herangehen motiviert werden. Wir geben allerdings zu bedenken, dass dieses Buch kein allumfassendes Kompendium zu dieser Thematik anbietet, sondern den Leser mit entsprechenden Verweisen weiterleitet.

Aufbau dieses Buchs

Die ersten drei Kapitel sind als Grundlagenkapitel zu verstehen. In Kapitel 1, *Einführung in die Arbeit mit Java*, widmen wir uns der Einrichtung der Arbeitsumgebung, der Versionsverwaltung und dem Build-Tool Maven. Hier liegt der Fokus auf der Software und den Tools. Gerade für Anwender, die nur gelegentlich mit Java arbeiten, ist dieses Kapitel dringend zu empfehlen. Dabei werden wir uns allerdings nicht mit den Tiefen von Maven beschäftigen, sondern eine praxisgerechte Schnelleinführung geben. Ziel ist es, alle Aufgaben in diesem Buch schnell und effizient lösen zu können.

Kapitel 2, *Java zum Auffrischen*, ist als Angebot an die Leser gedacht, die ihre Grundkenntnisse in der Programmierung auffrischen möchten oder Programmierkonzepte in Java gegebenenfalls nachschlagen wollen. Dazu werden einzelne Themen wie der Aufbau eines Programms, Bibliotheken, Variablen und Schleifen übersichtlich dargestellt. Beispiele und weitere Erklärungen bieten einen einfachen Einstieg.

Den Abschluss dieses Themenblocks bildet Kapitel 3, *Data Engineering mit Java*. Hier wird in die Grundprinzipien des Data Engineerings eingeführt, und mit vielen Beispielen werden Konzepte wie XML, JSON, API-Schnittstellen etc. dargestellt und ausgeführt. Hier finden sich auch erste Beispiele aus den Lebenswissenschaften. Auf diese einführenden Kapitel wird im Folgenden immer wieder verwiesen. Der erfahrene Leser kann direkt bei den späteren Kapiteln anfangen und bei Bedarf auf diese Kapitel zurückgreifen.

Sinnvoll erschien es uns, an dieser Stelle direkt das Thema *Data Mining* (Kapitel 4) anzuschließen. Hier beschreiben wir die klassischen Themen dieses Felds und insbesondere die Themen Klassifizierung und Clustering. In diesem Kapitel widmen wir uns primär den Grundlagen, da diese Methoden in fast allen darauffolgenden Kapiteln verwendet werden.

Die weiteren Kapitel behandeln konkrete Problemstellungen aus den Lebenswissenschaften. Zunächst wird auf die *Netzwerkanalyse: Graphen mit Java* in Kapitel 5 eingegangen. Dieses Thema erschien uns besonders wichtig, da Netzwerke sehr häufig verwendet, aber selten thematisch und technisch solide vorgestellt werden. Neben der technischen Einführung in die Bibliothek JGraphT geben wir hier viele Beispiele für gerichtete, ungerichtete oder auch andere Graphen und die Probleme, die damit gelöst werden können. Auch wenn wir den Versuch gewagt haben, die Themen möglichst breit aufzustellen, können wir keine allumfassende Einführung in dieses Thema geben. So können spezielle Aspekte wie phylogenetische Bäume hier nicht behandelt werden.

In Kapitel 6, *Bildverarbeitung mit Java und ImageJ*, findet sich eine umfangreiche Einführung in die Arbeit mit der Bibliothek ImageJ. Viele Probleme der Lebenswissenschaften beruhen auf Bilddaten, z. B. die Tumorerkennung oder die automatisierte Analyse von Pflanzenwuchs. Auch hier müssen wir auf viele teilweise sehr spezifische Aufgaben der Bildverarbeitung verzichten. So werden Sie vielleicht eine detaillierte Einführung in die KI-Methoden dieses Gebiets vermissen, auch wenn wir über die Verwendung von WEKA sprechen werden. Die folgenden Themen werden im Speziellen behandelt: die Partikelanalyse, die Objektklassifizierung und die Farbanalyse. Darüber hinaus runden zwei Praxisprojekte aus der Bildverarbeitung das Kapitel ab. Die Themenauswahl wurde insofern mit Bedacht gewählt, als dass dadurch alle wichtigen Methoden (und Fallstricke) von ImageJ als Bibliothek behandelt werden und der Leser sich sehr zügig weitere Felder erschließen kann.

Ein klassischer Themenbereich der Bioinformatik wird in Kapitel 7, *Sequenzanalyse mit BioJava*, aufgegriffen. Dieses Feld hat Anfang der 2000er-Jahre den Sprung zum *Next-Generation Sequencing* (NGS) gemacht und wird uns, da es die Bausteine des Lebens betrachtet, auch die nächsten Jahrzehnte beschäftigen. Es ist daher nicht nur ein wichtiger Bestandteil, sondern auch ein würdiger Abschluss unserer Themenauswahl. In der Einführung zu diesem Kapitel beschäftigen wir uns mit den programmiertechnischen Grundlagen der Sequenzanalyse mit BioJava und gehen auch auf die Datenbanksuche ein. Anschließend werden wir einzelne Themen wie »Multiple Sequence Alignment«, »BLAST« und »NGS« behandeln und in Übungsaufgaben vertiefen. Auch an dieser Stelle mussten wir die Themenauswahl sinnvoll einschränken; hierbei haben wir uns davon leiten lassen, welche Themen zügig und vor allem sinnvoll mit BioJava umgesetzt werden können. Für viele Spezialprobleme benötigen Sie dann weitere Bibliotheken oder Anwendungen. Diese können mit dem Wissen aus diesem Buch aber gut in eigene Workflows eingebettet werden.

Übungsaufgaben und Lösungen

Alle Kapitel dieses Buchs sind mit Übungsaufgaben versehen, in denen der Leser das Gelernte vertiefen und anwenden kann. Die Übungsaufgaben in den ersten zwei Kapiteln haben noch keinen Bezug zu den Lebenswissenschaften, oder der Bezug wird bestenfalls sehr künstlich hergestellt, da es sich hier um Grundlagen der Programmiersprache Java handelt. In allen folgenden Kapiteln werden vor allem Praxisprobleme aus den Lebenswissenschaften als Aufgaben verwendet. Wir stellen weiteres Material (wie Eingabedateien), Lösungsvorschläge, Lösungsanregungen und teilweise auch Lösungen in einem GitHub-Repository unter *https://github.com/jd-s/Java-fuer-die-Life-Sciences-Loesungen* zur Verfügung. Gern können Sie uns schreiben, wenn Sie eine elegantere, schnellere oder schönere Lösung zur Verfügung stellen wollen. Bei der Programmierung gibt es nicht eine einzige, sondern viele verschiedene Lösungen, die zum Ziel führen.

Weitere Informationsquellen

Dieses Buch ist selbstverständlich nicht das einzige Werk zu diesem Thema, auch wenn es unseres Wissens das einzige aktuelle deutschsprachige Werk ist. Wir haben für Sie verschiedene Literaturangaben und Verweise auf Internetressourcen zu einzelnen Themen zusammengestellt. Weitere Verweise finden Sie in den einzelnen Kapiteln in den Fußnoten.

Java im Allgemeinen:

- Christian Ullenboom: *Java ist auch eine Insel* (Rheinwerk 2020, die Ausgabe von 2017 ist auch als Internetressource verfügbar unter *http://openbook.rheinwerk-verlag.de/javainsel/*).
- Dirk Louis, Peter Müller: *Java – Der umfassende Programmierkurs* (O'Reilly 2014).
- Robert Liguori, Patricia Liguori: *Java – kurz & gut* (O'Reilly 2018).

Im Zusammenhang mit *Life Science Informatics und Java* sind uns nur zwei Veröffentlichungen bekannt:

- Peter Garst: *Mastering Java through Biology* (letzte Version von 2014; arbeitet mit Java 8).
- Harshawardhan Bal, Johnny Hujol: *Java for Bioinformatics and Biomedical Applications* (Springer 2010).

Bioinformatik und Life Science Informatics:

- Lee Harland, Mark Forster: *Open Source Software in Life Science Research: Practical Solutions to Common Challenges in the Pharmaceutical Industry and Beyond.* Woodhead Publishing Series in Biomedicine (Elsevier, 2012).
- Thomas Dandekar, Meik Kunz: *Bioinformatik: Ein einführendes Lehrbuch* (Springer 2017).
- Andrea Hansen: *Bioinformatik: Ein Leitfaden für Naturwissenschaftler* (Springer 2013).
- Martin Dugas, Karin Schmidt: *Medizinische Informatik und Bioinformatik: Ein Kompendium für Studium und Praxis* (Springer 2003).

Die in diesem Buch verwendeten Konventionen

In diesem Buch werden die folgenden typografischen Konventionen verwendet:

Kursiv
: Kennzeichnet neue Begriffe, URLs, E-Mail-Adressen, Dateinamen und Dateierweiterungen.

`Feste Zeichenbreite`
: Kennzeichnet Programmlistings sowie Programmelemente in Absätzen, wie etwa Variablen- oder Funktionsnamen, Datenbanken, Datentypen, Umgebungsvariablen, Anweisungen und Schlüsselwörter.

`Feste Zeichenbreite fett`
: Zeigt Befehle oder anderen Text an, der vom Benutzer wortwörtlich eingegeben werden soll.

`Feste Zeichenbreite kursiv`
: Kennzeichnet Text, der durch Werte ersetzt werden soll, die der Benutzer vorgibt oder die sich aus dem Kontext ergeben.

Dieses Symbol kennzeichnet einen Tipp oder Vorschlag.

Dieses Symbol zeigt eine allgemeine Bemerkung.

Dieses Element symbolisiert einen Warnhinweis.

Verwendung von Codebeispielen

Dieses Buch ist dazu gedacht, Ihnen bei Ihren Aufgaben zu helfen. Grundsätzlich dürfen Sie die mit diesem Buch bereitgestellten Codebeispiele in Ihren Programmen und der dazugehörigen Dokumentation nutzen. Sie müssen uns dazu nicht um Erlaubnis fragen, es sei denn, Sie reproduzieren einen beträchtlichen Teil des Codes. Schreiben Sie beispielsweise ein Programm, das mehrere Codeabschnitte aus diesem Buch enthält, benötigen Sie keine Erlaubnis. Verkaufen oder verteilen Sie dagegen eine CD-ROM mit Beispielen aus Büchern von O'Reilly, brauchen Sie eine Erlaubnis. Eine Frage mit einem Zitat aus diesem Buch unter Angabe eines Codebeispiels zu beantworten, benötigt keine Erlaubnis. Wollen Sie dagegen einen wesentlichen Anteil der Codebeispiele aus diesem Buch in Ihr eigenes Buch integrieren, brauchen Sie eine Erlaubnis.

Wir freuen uns über Zitate, verlangen diese aber nicht. Ein Zitat enthält üblicherweise Titel, Autor, Verlag und ISBN, zum Beispiel: »*Java für die Life Sciences: Eine Einführung in die angewandte Bioinformatik* von Jens Dörpinghaus, Sebastian Schaaf und Vera Weil (O'Reilly 2021), ISBN 978-3-96009-125-7«.

Wenn Sie glauben, dass Ihre Nutzung von Codebeispielen über das gewöhnliche Maß hinausgeht oder außerhalb der oben vorgestellten Nutzungsbedingungen liegt, kontaktieren Sie uns bitte unter *kommentar@oreilly.de*.

Danksagung

Wir danken unseren Gutachterinnen und Gutachtern, die den Entstehungs- und Reifeprozess dieses Buchs positiv begleitet haben. Besonders umfangreich und detailliert waren die Rückmeldungen von Christof Meigen, Heike Kattenbusch und Kristian Rother. Hervorheben möchten wir auch den positiven Einfluss vieler Studierender der letzten Jahre – deren Rückfragen, Kritik und teils überraschende Leistungen in der Bearbeitung von Übungsaufgaben und Softwareprojekten stellen letztlich das Fundament dieses Buchs dar. Für das geduldige und motivierende Lektorat unseres Erstlingswerks möchten wir uns bei Alexandra Follenius herzlich bedanken.