

Chapter 1

Introduction

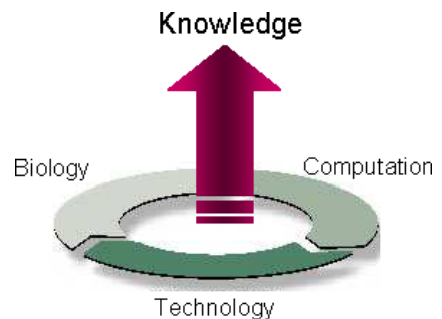
This chapter describes the motivation for this book, and sets out its aims and its organization.

The increasing role of computational analysis in biology

Over the past decade, research in molecular and cell biology has increasingly looked beyond reductionism and towards an integrated understanding of molecular and cellular systems. This is in part due to the virtuous cycle in which new technologies enable faster, cheaper, higher-resolution, and more comprehensive measurements (illustrated in the schematic on the right). The larger and more complex datasets that become available in this way are often too large to be analyzed manually.

Computational methods are needed first to preprocess the raw data, and then to extract meaning and insight from the data.

The widespread availability of computational resources is in turn creating opportunities for ever-more sophisticated experimental technologies, which lead to more data and new computational demands. An example of this feedback process in action is the emergence of microfluidic and other single-cell assays in recent years. The advent of microfluidic devices led to the development of computational tools both to control the devices and also to record, process, and interpret readings from them. Increasing automation has permitted multi-parameter measurements from large numbers of cells, which has in turn enabled statistical analysis of gene expression in single cells.

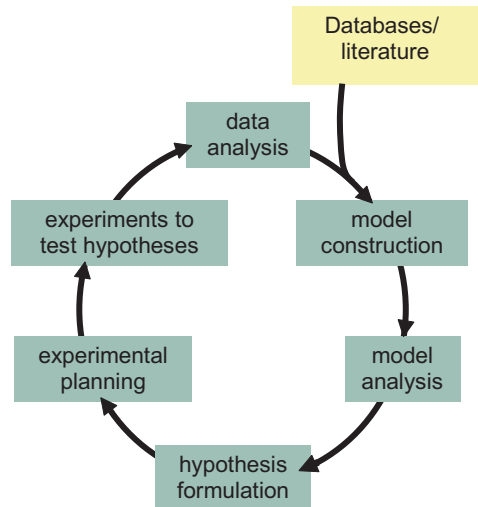


The payoff of the virtuous feedback cycle illustrated above is biological knowledge and insight. But the cycle also has an unplanned side-effect: more and more of the application of the scientific method in molecular and cell biology involves computers and computation. The scientific method (i.e. the process by which scientists, collectively and over time, construct more reliable and consistent representations of phenomena) can be described in four steps:

1. Careful observation of a system of interest;
2. Development of one or more hypotheses about the system observed;
3. Predictions made based on the proposed hypotheses; and
4. Performance of experiments that falsify or validate the predictions (and hence the hypotheses).

Biology is an experimental discipline, but good experiments are the result of considerable thinking, planning, and analysis. As experimental technologies — and associated data — become more complex, the necessary thinking, planning, and analyses are becoming ever more intellectually demanding. As a result, computational model building and model analysis have become integral parts of biology.

The figure at right shows the key steps in a typical molecular biology research project today. Computing supports every step of the process — even the experiments, which are typically controlled by embedded software (e.g. threshold detection in real-time PCR). In confining experimental work to a single step in this diagram, I do not mean to belittle its importance, but rather to emphasize the large number of steps that we typically go through at a desk rather than a bench. Traditionally, many of these steps were performed in the experimenter's head, and communicated via box-and-arrow diagrams. However, the size and complexity of the systems under study today increasingly require mathematics and computing to identify key features of the data, develop conceptual models, and formulate testable hypotheses.



Because of the intricacy of gene regulation, genetic regulatory networks (GRNs) — even networks of just two or three genes — can exhibit remarkably complex behaviors. Moreover, the construction of GRN models frequently involves the analysis of large volumes of data. As a result, research into GRNs can particularly benefit from computational model building and model analysis methods. Additionally, computational GRN models can be analyzed mathematically, explored interactively, dissected with *in silico* experiments, and communicated unambiguously (discussed in Chapter 4).

Molecular biology is in the early days of a trend towards increasing automation of experimental protocols. Manipulating and measuring gene regulation and expression are becoming easier, but the interpretation and analysis of the data generated increasingly require sophisticated computational tools. Experimental biologists of the future will need to have many of the skills of the computational biologists of today. The best molecular biologists of the future will be those who are not only excellent experimentalists, but also competent and effective users of computational tools and complex technologies. I hope that this book will provide a stepping stone in that direction for bench biologists. For theoreticians, I hope this book will provide a useful overview of the many different computational techniques used in the study of GRNs.

What this book tries to achieve

My aim throughout this book is to explain the “how” and “why” of modeling GRNs in simple language that I hope will be accessible to all readers irrespective of their educational background. I have also tried to keep the book as short as possible while still covering all of the fundamentals of GRN modeling theory. I hope that this brevity will make the book more suitable for teaching purposes, and less daunting for self-study.

The focus of this book is on the *regulation* of gene expression and on *networks* of interacting genes. There are a great many very interesting and related topics, such as genetics, gene discovery via sequence annotation, protein structure and function prediction, and modeling of signal transduction and metabolic networks, that are not explored in this book. There are several reasons for this. Firstly, there are already several good books on each of these topics. Secondly, a book covering all of these topics would be thousands of pages long, or

no more than a cursory survey. Last but not least, the computational techniques used in each of these disciplines are quite distinct.

The greatest use of GRN modeling in research is to provide new insights into GRN function and organization. I have made no attempt to present biological insights gained from computational studies of GRNs. This is partly because that would require a very different kind of book, but also because this is such a fast-moving topic that I suspect much of what I could cover would be out of date by the time the book is published. Instead, this book provides a broad introduction to the fundamentals of the many computational methodologies for building and analyzing models of GRNs.

Throughout the book, I assume that the reader is either already well versed in the biology of gene regulation and genetic regulatory networks, or is able and willing to learn these topics from the textbooks and resources I have listed in each chapter. There are two books that I would like to recommend to all readers here, since otherwise I would need to reference them repeatedly in every chapter:

- Eric Davidson's *The Regulatory Genome* (Academic Press, 2006). Although the book is focused on *developmental* gene regulatory systems in *animal* embryos, it is densely packed with insights and observations applicable to all GRNs. No serious student of GRNs can be without this book.
- Uri Alon's *An Introduction to Systems Biology — Design Principles of Biological Circuits* (Chapman & Hall/CRC, 2007). In order to focus on methods, I have studiously avoided discussing the biological insights arising from modeling GRNs. Alon's book is a beautiful demonstration of the value of modeling and the perfect companion to this book. I hope you will read the two books in parallel.

Who should read this book

Over the past decade, I have been asked repeatedly by students, collaborators, and colleagues what a traditionally educated experimentalist should read to understand the principles of computational biology. Needless to say, I never had a satisfactory answer.

I have written this book specifically with experimental biologists in mind. Experimentalists routinely use computational aids to design a primer sequence, process flow cytometry

data, or look for genes with similar sequences to their latest discovery. The topics presented in this book are a natural extension of these computational aids. As far as possible, I have tried to explain everything in intuitive terms, avoiding mathematical jargon. Where equations are unavoidable, I have tried to derive them from first principles, or at least provide an intuitive description.

The mathematicians, engineers, physicists, and computer scientists who are computational biologists today have typically spent many years learning the foundations of theoretical thinking in a way that no single book can summarize. This book is my attempt to provide a *primer*. I hope that interested readers will go on to take advanced courses in the specialist topics presented here in single chapters.

In addition to addressing the needs of experimentalists, I hope this book will prove useful to theoreticians new to computational biology. One of the biggest surprises for theoreticians moving into computational biology is that the range of theoretical frameworks and techniques that computational biology draws on is remarkably broad and growing. Researchers with degrees in engineering, mathematics, statistics, physics, computer science, etc. often have training in only a small fraction of the full range of techniques used in computational biology (e.g. algorithm design, logic, differential equations, statistical methods). This leads to the allegation that, for many theoreticians, “If the only tool you have is a hammer, every challenge looks like a nail.”

For theoreticians, this book can serve as an introduction to (some of) the many different computational approaches to GRN modeling and analysis. For readers interested in further detail, I have provided references to key publications and web sites.

There is little room within traditional university degree structures for a course on computational modeling of GRNs. However, in recent years, there has been increasing recognition of the need to move beyond archaic discipline boundaries. New departments and interdisciplinary degrees in bioengineering, genome sciences, and systems biology are examples of this trend. This book can form the basis of a semester-long introductory course within such degrees.

By focusing on a very specific biological topic, this book attempts to give nonspecialists a balanced grounding in the principles of computational modeling. I hope that the book will

prove useful both for teaching and also for self-study by anyone interested in integrated studies of genetic regulatory networks.

How this book is organized

The chapters of this book can be divided into three sections. The introductory section deals with the philosophical and conceptual infrastructure of modeling. Readers eager to start modeling may be tempted to skip these chapters. However, the concepts presented are crucial for correct application of modeling principles. I therefore urge anyone new to modeling not to skip this section.

The middle section, which forms the bulk of the book, is organized as chapters on distinct modeling frameworks. While these chapters can be read independently, I have organized them so as to start with a detailed biological picture and gradually introduce increasingly abstract perspectives. In this way, I hope the reader will be able to see the biological context of the theory-rich later chapters. The final section puts the preceding chapters in perspective, and highlights future attractions. I hope instructors will find the order of the book chapters a natural progression for classes.

Notwithstanding the above considerations, parts of this book may be more interesting to readers than others. For example, if they are engaged in research with a particular model organism, they may find some chapters of direct relevance, while others may not be applicable to their data. I have therefore tried to write this book in a way that (I hope) will allow the reader to dip in and start reading wherever they find something of interest. Cross-references within chapters guide readers to other relevant chapters in the book.

Example models are provided in the chapter appendices. With the exception of Matlab, I have used software tools that are freely available and easy to install and use by nonexperts. I urge readers to explore these models interactively. Instructors may find the models useful as starting points for laboratory exercises.

Throughout the book, I have provided URLs for software tools and other resources of relevance. Unfortunately, web addresses and page contents can change over time. If readers find that a URL listed in this book is out of date, they can use the Wayback Machine internet archive (<http://www.archive.org/web/web.php>) to retrieve earlier versions.

Extensive footnotes provide additional references and explanations. They allow the book to be read at two different levels. Readers interested in details can read the footnotes immediately, while others may elect to skip the footnotes at first reading. Because references are integrated into the main text in this way, no separate bibliography is provided.

Acknowledgments

I am deeply grateful to Eric Davidson and Ellen Rothenberg for their incisive comments on multiple iterations of many chapters of this book.

Many colleagues and collaborators generously read and commented on early versions of various chapters. In alphabetical order, Rod Adams, Uri Alon, Pedro de Atauri, Christophe Battail, Jim Collins, Constantin Georgescu, Pablo Iglesias, Bill Longabaugh, Kevin Murphy, Steve Ramsey, Mark Robinson, Alistair Rust, Maria Schilstra, Ilya Shmulevich, Kelly Smith, Mike Smoot, Yi Sun, Denis Thieffry, Vesteynn Thorsson, and Tau Mu Yi all generously read and corrected drafts. This book is infinitely better because of their suggestions and advice. I owe them all a huge debt of gratitude.

I would like to thank Wanda Tan, Lizzie Bennett, and Laurent Chaminade at Imperial College Press for their efficient and effective handling of my manuscript. They made the complex publishing process feel easy.

Finally, my partner Cecilia Bitz has helped me with the writing of this book in so many ways that she ought to be a coauthor. I want to thank her especially for rescuing me every time I found myself trapped in a mathematical quandary.

Feedback

If you find any errors in the book, or have suggestions for improvements, please email me (HBo1ouri@gmail.com) with the subject heading: Book Feedback. Thank you.